# COMPONENTE DE EXTRACCIÓN Y ALMACENAMIENTO DE DATOS DE UNA RED SOCIAL PARA UNA HERRAMIENTA WEB


**UNIVERSIDAD CATÓLICA DE COLOMBIA**

**FACULTAD DE INGENIERÍA**

**PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

**BOGOTÁ D.C**



**2017**

# COMPONENTE DE EXTRACCIÓN Y ALMACENAMIENTO DE DATOS DE UNA RED SOCIAL PARA UNA HERRAMIENTA WEB

**MILTON DANIEL REY SUÁREZ**

**Trabajo de Grado para Optar al Título de**

**Ingeniero de Sistemas**

**DIRECTOR**

**DIEGO ALBERTO RINCÓN YÁÑEZ MCSc**

**INGENIERO DE SISTEMAS**

**UNIVERSIDAD CATÓLICA DE COLOMBIA**

**FACULTAD DE INGENIERÍA**

**PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

**BOGOTÁ D.C**

**2017**

**NOTA DE ACEPTACIÓN**

Aprobado por el comité de grado en cumplimiento de los requisitos exigidos por la Facultad de Ingeniería y la Universidad Católica de Colombia para optar al título de Ingenieros de Sistemas.

_____

Jurado

_____

Diego Alberto Rincón
Director

_____

Revisor Metodológico.

Fecha de Entrega

## AGRADECIMIENTOS

Le doy gracias a mi director de trabajo de grado, Ing. Diego Alberto Rincón quien con su amplio conocimiento me orientó en el desarrollo del presente proyecto. A todos los docentes que hicieron parte de este proceso de formación por su tiempo y conocimiento.

De igual manera, expreso mis más sinceros agradecimientos a mi familia por su apoyo incondicional, de manera especial a mi madre por su gran amor, esfuerzo y dedicación, por todas sus enseñanzas, gracias mamá.

A todos mis amigos y compañeros de Universidad por las experiencias y vivencias compartidas.

Finalmente, a todas aquellas personas que contribuyeron de alguna manera en mi formación académica y profesional. A todos gracias.

TABLA DE CONTENIDO

## TABLA DE IMÁGENES

## TABLA DE ANEXOS

## ABSTRACT

This document describes the process through which the development of a tool that retrieves and stores information from the digital social network Facebook was achieved. The purpose of the tool called “Componente de extracción y almacenamiento” is to be an important source of information that could be statistically analyzed by an external software or future software developments in order to obtain predictive information that can be used in a specific context. The process through which the tool was developed includes the phases of analysis, design, development and tests, that is because it is necessary to be sure that the tool works correctly.

The process of the development of the tool was realized following a methodology called extreme programming for soloists, this methodology is focused on the development that is made by just one programmer. Following this methodology allowed the participants involved in this project to achieve the tests phase in which it was evidenced that the processes of retrieving and storing information were developed successfully.

The “Componente de extracción y almacenamiento” opens the possibility of developing new solutions that can be integrated with the software exposed in this project in order to make the tasks of analysis mentioned above. Finally, this document shows that the tool was developed using different technologies, according to this, it is concluded that is possible to interchange data and information between applications and different programming languages through web services, and this makes it possible to develop the functionalities of a specific software without limiting itself to a single language.



**Keywords: Retrieve information; Store information; Methodology phases; Extreme programming for soloists; Programming languages.**

## RESUMEN

El presente documento describe una herramienta que extrae y almacena datos e información de la red social Facebook, la cual servirá en un futuro como una importante fuente de datos en formato estándar sobre los cuales se podrá realizar análisis estadístico enfocado en algún contexto especifico. En el presente proyecto, los desarrollos de las funcionalidades de la herramienta se aplican en el contexto social de violencia política en redes sociales, específicamente en la red social Facebook. A esta herramienta se le llamará en el presente documento “Componente de extracción y almacenamiento”.

El proceso de desarrollo del proyecto está dado por fases de análisis, diseño, implementación y pruebas en busca del funcionamiento óptimo del componente, acompañado de la adaptación de tecnologías existentes para el desarrollo de software, las cuales permiten definir el diseño del componente y consolidar los conceptos en la implementación general, adicionalmente este proceso se realizó siguiendo la metodología de desarrollo de programación extrema para solistas, la cual está enfocada al desarrollo por parte de un solo programador y permitió llegar hasta las fases finales de pruebas y resultados en las cuales se evidencio que la información es extraída y almacenada de forma ágil y sin alterar la integridad de los datos.

Con la construcción del componente de extracción y almacenamiento se abre la posibilidad de desarrollar e integrar múltiples soluciones al presente proyecto con el objetivo de realizar las tareas de análisis mencionado anteriormente. En relación a lo anterior, con el uso de distintas tecnologías empleadas en este proyecto se concluye que es posible intercambiar información entre aplicaciones y lenguajes de programación mediante el uso de servicios web permitiendo de esta manera hacer una integración sin limitar las distintas funcionalidades a una sola tecnología.



**Palabras Clave: Extracción y almacenamiento de información; red social; metodología; lenguajes de programación.**

# INTRODUCCIÓN

Internet se convirtió en la principal puerta de acceso al conocimiento, entretenimiento e información alrededor del mundo (Howard, Rainie, & Jones, 2001), los avances de las tecnologías de la información y de las comunicaciones han incorporado nuevas herramientas y formas de intermediación e interactividad que están reconfigurando la forma en la que las personas comparten sus pensamientos y se comunican entre sí. En adición, el fenómeno mediático actual de las redes sociales es la consecuencia de la evolución de internet, aunque inicialmente tenían fines netamente de entretenimiento con el paso de los años estas redes sociales han ido creciendo y ampliándose hasta convertirse en algo más serio y organizado(Howard et al., 2001).

Cabe agregar que el uso de redes sociales digitales está generando constantemente un volumen enorme de información, datos y metadatos que crece todos los días y sobre los cuales se puede realizar algún tipo de análisis para obtener beneficios estratégicos de estos. Sin embargo, estos datos están contenidos en plataformas como foros de opinión, blogs, wikis etc. Haciendo difícil la tarea de análisis y tratamiento de la información, debido a la variedad de datos e información en diferentes formatos que puede estar allí contenida.

En referencia a lo anterior, las cuentas asociadas a personajes públicos en diferentes contextos (económico, político, social, religioso, espectáculo etc.) tienen un mayor número de seguidores, lo que supone más población con acceso a la información que éstas publican, lo que supone que las personas puedan conocer la posición o el pensamiento de dichos personajes públicos frente a una situación y estas posiciones o inclinaciones se ven plasmadas por medio de los estados o publicaciones en sus respectivas cuentas; y es precisamente allí en donde se desarrolla el contexto del presente proyecto, el cual pretende facilitar la medición de la violencia política en Colombia expresada en la red social Facebook mediante un componente que permita extraer y almacenar información para un posterior análisis o medición.

Este documento presenta de manera formal, el proceso de desarrollo de un componente que permite realizar la extracción de datos de la red social Facebook y posteriormente almacenarlos, permitiendo, tener información sobre la cual se pueda realizar algún tipo de análisis. Cabe resaltar que el desarrollo de este proyecto está articulado con el desarrollo de una plataforma analítica en el marco de un proyecto de investigación asociado a la facultad de ingeniería de la Universidad Católica de Colombia.

# 1. GENERALIDADES

## 1.1 ANTECEDENTES

El concepto de redes sociales no es un tema que se pueda considerar como nuevo, de hecho, las primeras aproximaciones al termino datan de comienzos del siglo XX con el sociólogo alemán Georg Simmel(Breiger, 2004), sin embargo, con la llegada de nuevas tecnologías, el acercamiento al concepto de red social se ha hecho realidad mediante la implementación de plataformas y páginas web.

Como consecuencia al auge de las redes sociales, resultó también, el análisis del comportamiento de los usuarios de estas plataformas por medio de las ideas que expresan o el contenido que comparten.

No obstante, para poder realizar la tarea de análisis, debe existir una manera de acceder a la información allí contenida, por tal motivo ha sido necesario el desarrollo de herramientas tecnológicas que permitan extraer información de páginas y plataformas web de tal manera que facilite el tratamiento de los datos para quien hace la labor de análisis.

Por medio del proceso de investigación adjunto al presente proyecto se presenta a continuación diferentes antecedentes de herramientas o desarrollos enfocados principalmente a la recolección de datos, algunas de estas herramientas son:

- ✓ Netvizz**:** Es una herramienta de colección de datos y extracción(Rieder, 2013) que permite a los investigadores exportar datos en formatos de archivo estándar de diferentes secciones del servicio de la red social Facebook. Las redes de amistad, grupos y páginas pueden ser analizados cuantitativamente y cualitativamente en lo que respecta a las características demográficas y características relacionales.

  Analiza los aspectos de la extracción de datos a través de la aplicación oficial interfaz de programación, y se ocupa brevemente de la ética vinculada a este tipo de investigación.

- ✓ Tsimmis**:** Es un componente para sistemas semiestructurados de gestión de datos, TSIMMIS (Chawathe et al., 1994) puede configurarse a través de archivos de especificaciones de usuario.

  Los archivos de especificaciones están compuestos por una secuencia de comandos que definen los pasos de extracción. Cada comando es de la forma [variables, fuente, pattern] donde las variables representan un conjunto que contiene los resultados de la extracción. Fuente, especifica el documento

de entrada que se debe considerar (por ejemplo, una página Web), y Pattern permite hacer coincidir los datos de interés dentro de la fuente. Los datos almacenados en las variables pueden ser utilizados como entrada para comandos posteriores.

El extractor está basado en un archivo de especificación que analiza páginas HTML para localizar los datos interesantes y extraerlos. Después del último comando se ejecuta el conjunto de variables que contienen los datos.

- ✓ Rapier : Producción Robusta Automatizada de Extracción de Información (da Silva & Teixeira, s. f.) es una herramienta destinada a extraer datos de texto libre. Toma como entrada una fuente que puede ser una página web o un documento y una plantilla que indica los datos a extraer. Esta plantilla se utiliza para aprender los patrones de extracción de datos.

  El algoritmo de aprendizaje incorpora técnicas de varios programas de lógica inductiva y aprende patrones ilimitados que incluyen restricciones sobre las palabras. Estos patrones consisten en tres espacios distintos: El Pro-, Post- y el relleno. Los dos primeros desempeñan el papel de delimitadores de izquierda y derecha en la oración, mientras que el ultimo describe la estructura de los datos a extraer. Rapier extrae un solo registro de cada documento tomado como entrada, y por lo tanto se denomina para ser "single-slot".

- ✓ Polyphonet: Es una herramienta que emplea varias técnicas avanzadas para extraer relaciones de personas en una red social, detectar grupos de personas y obtener palabras clave para una persona. Los motores de búsqueda, especialmente Google, se utilizan para medir la co-ocurrencia de la información y obtener documentos Web.

  Polyphonet (Matsuo et al., 2007), se ha utilizado en cuatro conferencias académicas, cada una con más de 500 participantes, propone una arquitectura novel llamada “Iterative Social Network Mining”. Utiliza módulos simples utilizando Google y se caracteriza por procesos de escalabilidad y relacionar-identificar: se repite la identificación de cada entidad y la extracción de relaciones para obtener una red social más precisa.

## 1.2 PLANTEAMIENTO DEL PROBLEMA

La violencia política puede considerarse como la acción de dirigirse contra los opositores ideológicos, ya sea para silenciar su opinión en contra el sistema, o para atentar contra la política que se ejerce(González, Bolivar, & Vázquez, 2003). Con referencia a lo anterior, los principales actores del conflicto en Colombia, tanto el gobierno desde sus diferentes entidades, como los diferentes grupos armados, expresan sus ideologías, posiciones y pensamientos a través de diferentes medios, uno de ellos es por redes sociales, porque es un medio al que muchas personas tienen acceso y además se ha convertido en un medio de opinión pública muy efectivo, tanto así, que es común ver manifestaciones de violencia política en distintas redes sociales digitales entre personajes públicos involucrados en la política del país.

Por otra parte, antes internet era un medio en el que las personas podían consultar información sin mayor interacción, sin embargo, con la llegada de la web 2.0 (Nafria, 2007) cuyo principal aporte fue brindar la posibilidad de subir y compartir información desde cualquier lugar, el uso de internet se hizo cada vez más frecuente, dando así paso al crecimiento de las redes sociales y sus diferentes usos. Para poner en contexto la influencia que tienen las redes sociales en aspectos políticos se cita un ejemplo, la campaña de Barack Obama a la presidencia en el año 2008 empleó diferentes redes sociales como Facebook y Myspace de una manera nunca vista a la fecha en una campaña electoral. La página de Facebook atrajo a tres millones de personas. Además, se colgaron más de 2.000 vídeos de YouTube que fueron vistos más de 15 millones de veces (Polo, s. f.). Un grupo de investigación del Massachusetts Institute of Technology, extrajo datos en plataformas web para mostrar las correlaciones entre la cantidad de medios sociales utilizados por los candidatos y el ganador de la campaña presidencial de 2008 (Gloor, Krauss, Nann, Fischbach, & Schoder, 2009). Este poderoso ejemplo hace énfasis en el potencial de los datos de las redes sociales para predecir resultados a nivel político, pero puede ser utilizado para generar información predictiva en otros campos.

Resulta oportuno mencionar que para el desarrollo del presente proyecto se presenta una problemática política y social referente a personas expuestas a situaciones traumáticas en un contexto de violencia política en Colombia, donde se busca obtener una solución por medio del análisis de información contenida en las cuentas de la red social Facebook de personajes públicos asociados a violencia política en Colombia, a través de la interacción de diferentes áreas del conocimiento que permitan llevar a cabo estas labores. Desde las áreas de ingeniería de sistemas y psicología se busca la aplicación de sus diferentes ramas del conocimiento para alcanzar una solución apoyada en tecnología que permita generar bienestar social y disminuir las secuelas y daños causados por la violencia política en el país.

Para ilustrar lo anterior, como es citado en el proyecto (EUROPSIS & GISIC, 2016) “La facultad de psicología realizará un estudio cuantitativo con dos fases. La primera, tiene como propósito identificar los estilos lingüísticos asociados al reconocimiento del conflicto, situaciones traumáticas y creencias identitarias en el contexto de violencia política en Colombia mediante el análisis estadístico de los comunicados conjuntos difundidos por la mesa de conversaciones en el marco del proceso de diálogo y negociación entre las Fuerzas Armadas Revolucionarias de Colombia y el Gobierno colombiano y las conversaciones de distintos actores del contexto político colombiano durante el año 2016 mediante mensajes textuales publicados en redes sociales”. Es precisamente en esta fase donde la ingeniería de sistemas participa de la investigación por medio del desarrollo de un componente de extracción y almacenamiento de datos de la red socia Facebook, con el cual se podrá recolectar información sobre la cual se realizará el análisis estadístico mencionado anteriormente. Cabe resaltar que, aunque el desarrollo del componente del presente proyecto se hace bajo un contexto de violencia política, éste podría ser enfocado en cualquier otro contexto, campo del conocimiento o industria en el que sea útil su funcionalidad. Con el objetivo de obtener y producir información valiosa de carácter predictivo que ayude a la toma de decisiones estratégicas en cualquier otro marco de referencia o contexto.

El desarrollo del presente proyecto expone una solución tecnológica que permite la recuperación y almacenamiento ágil de información contenida en una red social, de los actores y bandos participes del conflicto y violencia política en el país, para que pueda ser analizada posteriormente. Dadas las condiciones que anteceden, es muy importante velar porque la información que vaya a ser analizada sea verídica, es decir, que la integridad de los datos no se altere al momento de ser extraídos, almacenados y analizados.

Las herramientas y soluciones gratuitas o de bajo costo del mercado tienen unas características y funcionalidades reducidas que no permiten realizar un correcto proceso de extracción y almacenamiento de la información, en algunos casos, los datos no tienen el formato o la calidad adecuada para que se haga un análisis correcto.

En ese orden de ideas, se ratifica la formulación y desarrollo de la herramienta expuesta en este proyecto que permite extraer de forma sencilla los datos y metadatos de las páginas de una red social, que se pueden utilizar en el análisis de diferentes contextos específicos dependiendo el enfoque que se requiera.

Dado el escenario de postconflicto armado en Colombia se requiere el desarrollo de estrategias y proyectos que asistan al bienestar de las comunidades, grupos o personas que han sido golpeados por la violencia armada y política. Lo anterior, por medio de desarrollo de software y aplicaciones que permitan innovar tecnológicamente frente a las aplicaciones tradicionales de análisis y medición.

## 1.3 FORMULACIÓN DEL PROBLEMA

La cantidad de información y datos digitales generados ha ido creciendo a lo largo de los años, al enviar correos electrónicos por e-mail o mensajes por aplicaciones de mensajería instantánea, publicar estados en una red social, compartir contenidos multimedia o responder a una encuesta, se crean nuevos datos que pueden ser analizados. Esto ha provocado cambios en la forma como se expresan conceptos clásicos como la opinión pública o democracia deliberativa, porque ahora se hace por medios en los que antes no se hacía, como el internet. De tal manera que los medios de comunicación en línea se han convertido en un espacio importante de socialización ciudadana y han generado un nuevo estilo de politización mediante la interconexión de personas y grupos que, por medio de diversos blogs personales, foros de opinión, portales de información y redes sociales deliberan y comparten hechos frente a realidades políticas.

Dadas las condiciones que anteceden y teniendo en cuenta los antecedentes presentados en este documento, existen en el mercado herramientas que hacen procesos de recolección de datos en contextos muy específicos, adicionalmente, las herramientas gratuitas o de bajo costo ofrecen funcionalidades muy limitadas las cuales recolectan información, pero los datos que se extraen no tienen la calidad necesaria para poder realizar un proceso analítico sobre ellos. Es por esta razón que se identifica un problema en el que no se cuenta con una solución tecnológica que sea capaz de adaptarse a diversos contextos, ofreciendo las funcionalidades de extracción y almacenamiento ágil de datos. Adicionalmente, en referencia a las consideraciones expuestas en el planteamiento general del problema, se requiere una herramienta o un medio que permita adaptarse a un contexto político-social en busca de obtener información procedente de redes sociales sobre la cual se pueda aplicar análisis estadísticos, sin embargo, se desconoce de una herramienta o medio que cumpla con las características y necesidades planteadas.

En síntesis, tomando como base todo lo citado anteriormente, para realizar el análisis correspondiente a los datos y metadatos generados en redes sociales es necesario recolectar y almacenar información para su posterior tratamiento. ¿A través de que medio o herramienta se puede recolectar y almacenar información procedente de redes sociales de forma ágil para realizar un futuro proceso de análisis estadístico?

## 1.4 JUSTIFICACIÓN

La facultad de Ingeniería de Sistemas junto a la facultad de Psicología de la Universidad encontraron que los estudios de legitimación sobre violencia política muestran que los grupos que perpetran acciones violentas necesariamente

construyen justificaciones ideológicas cuyos contenidos pretenden minimizar el impacto emocional negativo que generan en los seres humanos y se expresan mediante discursos fundamentados en el reconocimiento del conflicto, la diferenciación grupal y el uso de creencias legitimadoras y deslegitimadoras de los actores en correspondencia con los ataques individuales o colectivos que en el contexto político usualmente se dirigen a favor o en contra del Estado y se constituyen en antecedente de experiencias traumáticas que causan consecuencias psicológicas y psicosociales a las personas y comunidades, en este caso, de la sociedad colombiana.

Como se mencionó anteriormente, la violencia política puede considerarse como la acción de dirigirse contra los opositores ideológicos, ya sea para silenciar su opinión en contra el sistema, o para atentar contra la política que se ejerce(González, Bolivar, & Vázquez, 2003).

Para poder evaluar que el método de narración escrita sobre experiencias traumáticas genera mejorías en los niveles de sintomatología de estrés psicosocial y salud auto percibido de las personas, es necesario hacer la extracción de datos de redes sociales con el objetivo de realizar una comparación de resultados entre el método tradicional y el método apoyado por la herramienta web.

El principal aporte que genera el desarrollo de esta herramienta es la posibilidad de generar soluciones a necesidades en distintos campos del conocimiento y la industria por medio del análisis aplicado a los datos extraídos de una red social y que permiten obtener información clave que sirve de carácter predictivo a la hora de tomar de decisiones estratégicas. Con los resultados de la investigación adjunta al presente proyecto se busca afectar de manera positiva adultos expuestos a situaciones traumáticas en un contexto de violencia política.

## 1.5 DELIMITACIÓN

### 1.5.1 Alcance

- ✓ El desarrollo del componente del presente proyecto permite la extracción limpia y el almacenamiento ágil de publicaciones o posts de páginas de carácter público asociadas a violencia política de la red social Facebook de personajes asociados a política, medios de comunicación, partidos políticos, grupos armados, entre otros.

### 1.5.2 Espacio

- ✓ El presente proyecto de desarrollo se lleva a cabo en la carrera de Ingeniería de Sistemas de la Universidad Católica de Colombia.

### 1.5.3 Tiempo

- ✓ La implementación del componente de extracción y almacenamiento de datos comprende un periodo de tiempo no mayor a 5 meses, periodo comprendido entre junio de 2017 y noviembre de 2017.

### 1.5.4 Contenido

- ✓ El desarrollo del componente permite la extracción limpia de los datos y metadatos de cuentas de carácter público de la red social Facebook, explotando las ventajas de obtención de información por medio del API proveído por la comunidad de desarrolladores de Facebook llamada “Graph API”.

- ✓ La herramienta permite un proceso de almacenamiento ágil gracias a la adaptación de conceptos de bases de datos, y el estudio sobre tecnologías adecuadas para la implementación de la solución.

- ✓ El almacenamiento de la información extraída se realizará en el sistema de base de datos NoSQL MongoDB alojado en un servidor de datos.

### 1.5.5 Limitaciones

- ✓ La actualización o retroalimentación de la herramienta se ve comprometida por el tiempo definido.

- ✓ La infraestructura o vista de despliegue comprometida en la implementación es básica.
- ✓ El desarrollo, implementación y documentación de este componente será ejecutado por una sola persona.

# 2. OBJETIVOS

## 2.1 OBJETIVO GENERAL

Implementar un componente que permita extraer y almacenar datos e información de cuentas públicas de la red social Facebook asociadas al contexto de violencia política en Colombia, mediante el uso de tecnologías abiertas.

## 2.2 OBJETIVOS ESPECÍFICOS

- ✓ Realizar un estado del arte sobre las de técnicas de recolección y extracción de datos conocidas como Information retrieval y Data retrieval.
- ✓ Construir un documento de especificación de requerimientos y diseño de un componente de extracción y almacenamiento de datos de la red social Facebook, documentos sobre los cuales se basará la implementación de la solución.
- ✓ Implementar un componente de extracción y almacenamiento de datos de la red social Facebook soportándose en tecnologías abiertas.
- ✓ Realizar las pruebas unitarias correspondientes que permitan garantizar el adecuado funcionamiento del componente desarrollado.

# 3. MARCO DE REFERENCIA

## 3.1 ESTADO DEL ARTE

La necesidad de almacenar y recuperar información se convirtió con el paso del tiempo en una tarea muy importante, especialmente con la invención del papel y posteriormente la imprenta. En la década de 1940 con el acercamiento a los primeros computadores, se pensó en la idea de almacenar grandes cantidades de información de manera mecánica(Singhal, 2001). Sin embargo, la ventaja del uso de computadores dio paso a la necesidad de encontrar y filtrar información útil de entre toda la cantidad de información guardada. El campo de la recuperación de información nació en la década de 1950 como respuesta a esta necesidad, este campo ha tenido una evolución considerable desde esas épocas hasta la actualidad, un claro ejemplo de esto son las tecnologías utilizadas por las personas diariamente como los motores web de búsqueda. La importancia alrededor de la recuperación de la información fue creciendo al punto de crear en 1992 una serie de conferencias anuales llamadas Text Retrieval Conference TREC(Technology, 2017b) en la que se reunían investigadores para exponer avances y fomentar nuevos modelos que permitían el desarrollo de este campo, dichas conferencias siguen en pie en la actualidad patrocinadas por National Institute of Standards and Technology(Technology, 2017a).

Information Retrieval (IR) (Han, Pei, & Kamber, 2011) o Recuperación de Información, es el área del conocimiento que se encarga de recuperar documentos electrónicos o información de carácter digital contenida en documentos generalmente de naturaleza no estructurada, estos documentos pueden ser texto o multimedia que comprende imagen, video, audio, animación, voz, gráficos, entre otros, y pueden estar contenidos en sistemas de información, bases de datos o la web. Por otra parte, la recuperación de datos ha permitido el avance en otros campos y ramas del conocimiento que van muy ligados a éste como la minería de datos(Han et al., 2011). Para ello existen modelos definidos que permiten la implementación de técnicas y algoritmos que permiten realizar extracción de datos. Los modelos definidos(Singhal, 2001) son el booleano, espacio vectorial, probabilístico, difuso y de imágenes. Estos han sido estudiados en detalle y han tenido implementaciones tanto para la experimentación, como con fines comerciales. Por ejemplo, una técnica aplicada a la recuperación de información en la web es realizar búsquedas por palabras clave, este es un método simple pero efectivo para acceder a datos estructurados. Dado que muchos conjuntos de datos

de la vida real están estructurados en tablas, árboles y grafos, la búsqueda de palabras clave sobre dichos datos o estructuras de datos se ha vuelto cada vez más importante y ha atraído mucho interés de investigación tanto en las bases de datos como en Information Retrieval(Aggarwal, 2011). En ese mismo sentido, existen otras técnicas que permiten llevar a cabo la recuperación de información, la clasificación de documentos es una aplicación clave para Information Retrieval y pretende agrupar los documentos según categorías o similitud en contenido. El procesamiento del lenguaje natural también se ha propuesto como una herramienta para mejorar la efectividad de la recuperación, pero ha tenido un éxito muy limitado debido a su nivel de complejidad(Liddy et al., 2000).

Dadas las condiciones que anteceden, la recuperación de información puede aplicarse sobre documentos contenidos en la web, y es allí donde entran las redes sociales a jugar un papel importante. En general, una red social se define como una red de interacciones o relaciones, donde los nodos están compuestos por actores, y los bordes consisten en las relaciones o interacciones entre estos actores(Aggarwal, 2011).

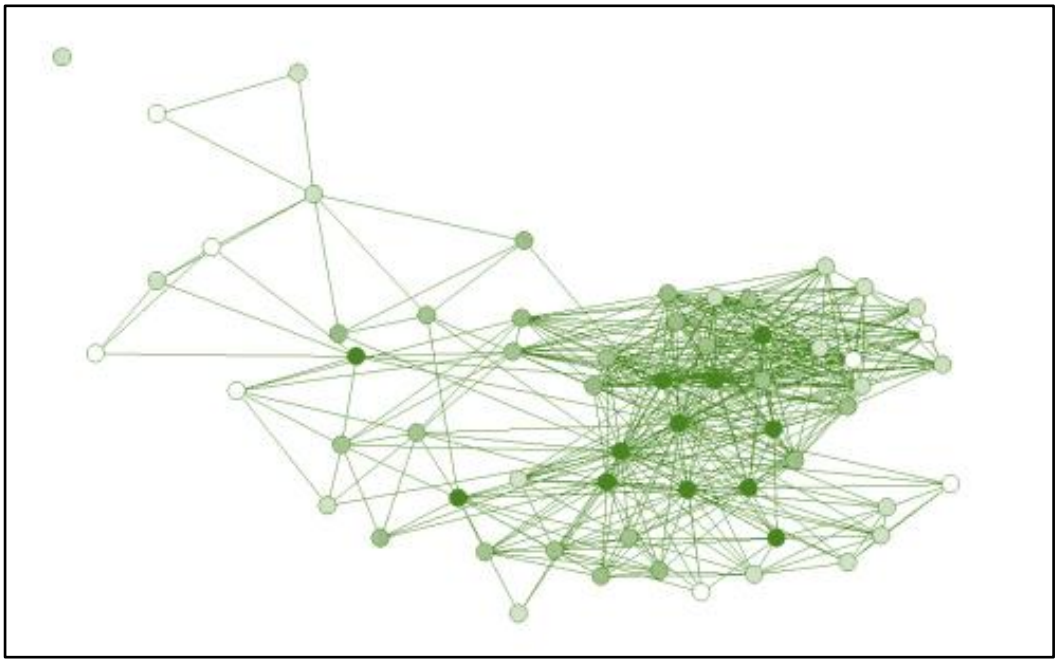
Cabe resaltar que el concepto de redes sociales no está restringido al caso específico de una red social basada en internet como Facebook, al contrario, el concepto de las redes sociales se ha estudiado a menudo en el campo de la sociología en términos de interacciones genéricas entre cualquier grupo de actores o personas. Estas interacciones pueden ser de cualquier forma convencional o no convencional, ya sean interacciones cara a cara, interacciones de telecomunicación, interacciones de correo electrónico o de otro tipo.

Por otra parte, las redes sociales pueden considerarse como un grafo, siendo los nodos los usuarios del sistema y las aristas la relación que existe entre ellos (Ver Figura 1). Cada nodo puede contener grandes cantidades de datos de texto y multimedia. No obstante, muchas páginas informales que se asemejan a redes sociales, como los blogs o plataformas web, también contienen una grandes cantidades de información en diferentes formatos. Muchos de estos sitios no tienen restricciones de privacidad para implementar un proceso de búsqueda efectivo, y en ese orden de ideas se pueden ejecutar técnicas y algoritmos de recolección de información sobre dichos sitios o plataformas.

No obstante, un sistema de recuperación de información no informa, es decir que cambia el conocimiento del usuario sobre el tema que se está consultando. Simplemente informa sobre la existencia sí la hay, y el paradero de documentos

relacionados con su solicitud(Mogotsi, 2010). La explosión y sobrecarga de información se han convertido en tema cliché en la actualidad, de allí la importancia de Information Retrieval como área de actividad, nos solo con fines comerciales sino también académicos.

**Ilustración 1.** Grafo de Teoría de Redes Sociales.



Fuente: Molina, J. L. (2004). La ciencia de las redes. *Apuntes de Ciencia y Tecnología*, *11*(1), 36-42.

Particularmente muchos desarrollos en el campo de Information Retrieval aplicado a las redes sociales digitales ha sido enfocado en “Social Network Data Analytics”(Aggarwal, 2011) debido al estrecho lazo que existe entre recuperar información de páginas web, y darle algún tipo de uso o tratamiento que generalmente deriva en análisis. El Departamento de Tecnología y ciencias de la computación de la Universidad de Tsinghua en Beijin realizó la aproximación de un (Tang, Zhang, & Yao, 2007)sistema de extracción de redes sociales de investigadores académicos basado en Information Retrieval en el año 2007, consiste en la extracción de la información de perfiles de investigadores que estaban adscritos a la red social. El sistema basa su funcionalidad de extracción en dos pasos que se describen a continuación. El primer paso es el procesamiento, en él se separa el texto en cinco tipos de palabras, teniendo las siguientes categorías (palabra estándar, palabra especial, <imagen>, término, signo de puntuación). Las palabras estándar son palabras en lenguaje natural(Liddy et al., 2000). Las palabras

especiales incluyen correo electrónico, URL, fecha, número, porcentaje, palabras que contienen símbolos especiales, entre otras. Las <imagen> son etiquetas de imágenes en el archivo HTML de las páginas consultadas. Los términos son frases básicas extraídas de las páginas web. Una vez realizado el primer paso de procesamiento se realiza el segundo paso dentro del proceso de extracción que es el etiquetado, en él se asignan posibles etiquetas a cada tipo de palabra, por ejemplo, para palabras especiales, se asignan etiquetas que pueden ser posición, afiliación, correo electrónico, dirección, teléfono, fax y Bsdate, Msdate y Phddate que corresponden a posibles campos dentro de la página web que contienen esta información. Finalmente, se almacenan todos los datos que fueron anteriormente tratados.

Como puede observarse, el campo de Data Retrieval o Information Retrieval ha evolucionado de la mano con las tecnologías y los estándares que hacen posible su funcionamiento, pero no se puede dejar a un lado un concepto importante en el manejo de grandes volúmenes de información como lo es Big Data. Según lo cita Amir Gandomi de la Universidad de Ryerson en su artículo “Beyond the hype: Big data concepts, methods, and analytics”(Gandomi & Haider, 2015), Big Data es un término que describe grandes volúmenes de datos de alta velocidad, complejos y variables que requieren técnicas y tecnologías avanzadas para permitir la captura, el almacenamiento, la distribución, la administración y el análisis de la información. La cantidad de información que comprende big data no toma valor si no se analiza. Su valor potencial se muestra cuando se aprovecha para impulsar la toma de decisiones. Para permitir tal toma de decisiones basada en evidencia, las organizaciones necesitan procesos eficientes para convertir grandes volúmenes de datos diversos en ideas significativas. Como se menciona en [21] el proceso macro para la obtención de información de valor del big data tiene asociado dos procesos generales que son gestión de datos y análisis de datos(Michael & Miller, 2013). La gestión de datos implica tecnologías de soporten adquirir, almacenar, preparar y recuperar datos para su análisis. Seguidamente el análisis se refiere a las técnicas utilizadas para analizar y adquirir inteligencia a partir de big data. Por lo tanto, el análisis de big data se puede ver como un subproceso en el proceso general de "extracción de información" de Big Data.

El análisis de la información se puede ramificar según el formato en el que se encuentre, el análisis de texto o data mining(Han et al., 2011) hace referencia a técnicas que permiten extraer información en formato texto que está contenida en los correos electrónicos, blogs, foros en línea, respuestas a encuestas, feeds de redes sociales, documentos corporativos, noticias, entre otros. El análisis de texto

permite a las empresas convertir grandes volúmenes de texto generado por humanos en resúmenes significativos, que respaldan la toma de decisiones basada en evidencia.

Las características más relevantes que diferencian la recuperación de información de la web y los sistemas de bases de datos tradicionales son dos(Han et al., 2011), la primera es que IR asume que la información y los datos bajo búsqueda en la web están sin estructurar y la segunda es que las consultas o “queries” están formadas principalmente por palabras claves que no tienen estructuras complejas, a diferencia por ejemplo de las consultas SQL (ver Figura 2) en los sistemas de bases de datos tradicionales.

El volumen cada vez mayor de información de texto y datos multimedia se ha puesto en línea a disposición de las personas debido al rápido crecimiento de la web y aplicaciones tales como bibliotecas digitales o gobiernos en línea. Por lo tanto, las técnicas de minería de textos y la minería de datos multimedia, integrados con métodos de recuperación de información IR, se han vuelto cada vez más necesarios(Han et al., 2011).

Los sistemas de bases de datos centran su atención en la creación, mantenimiento y uso de bases de datos para organizaciones y usuarios finales. De manera particular estos sistemas han establecido unos principios altamente reconocidos en modelos de datos, lenguajes de consultas, almacenamiento de datos, indexación y métodos de acceso a la información(Lotfy, Saleh, El-Ghareeb, & Ali, 2016).

No obstante, las bases de datos relacionales pretenden que todos los datos estén estructurados y tipados, generando dependencia de la información contenida en tablas. Actualmente las aplicaciones y los sistemas de información buscan atender millones de peticiones de clientes en muy poco tiempo sin dejar a un lado aspectos de calidad como tiempos de respuesta, disponibilidad e integridad de la información(Parker, 1987) , al tratarse de grandes cantidades de datos, estas bases deben estar soportadas en múltiples servidores que no necesariamente se encuentran ubicados en el mismo espacio geográfico sin perder las relaciones entre los datos.

Existen otras alternativas como las bases de datos NewSQL(Melton & Simon, 1993) y las no relacionales NoSQL, estas no establecen estrictas relaciones entre los datos permitiendo el almacenamiento de volúmenes muy grandes de datos y metadatos, este tipo de bases de datos son utilizadas usualmente en la práctica de

Big Data. Las bases de datos no relacionales también son conocidas como NoSQL (Lotfy et al., 2016), en referencia a lo citado anteriormente SQL es un lenguaje de consultas y está formado por estructuras predefinidas que pretenden definir, manipular y controlar los datos. En el caso de las bases de datos no relacionales soportan el uso del lenguaje de consultas SQL, sin embargo, no es muy común ver este lenguaje en esas bases de datos porque precisamente existen otras maneras de consultar dichos datos.

**Ilustración 2.** Ejemplo de la sintaxis de una consulta de actualización en SQL.

```
UPDATE table_name

SET column1 = value1, column2 = value2,... columnN = valueN

WHERE [condition];
```

Fuente: Elaboración propia.

Las bases de datos NoSQL hacen referencia a todas aquellas clases de sistemas de gestión de bases de datos que no se adhieren al tradicional modelo de gestión de bases de datos relacionales, que se caracteriza por almacenar la información en tablas utilizando el lenguaje SQL para la manipulación de datos. Por otra parte las bases de datos no relacionales no siguen las características ACID(Lotfy et al., 2016) propias de las bases de datos relacionales, estas buscan que la información sea durable, atómica, consistente y asilada, procurando por la integridad de la información sobre otros aspectos como el rendimiento o la escalabilidad de la base de datos. En referencia a lo anterior, las bases de datos NoSQL se aproximan a un modelo llamado BASE(Vaish, 2013) que consiste en brindar una disponibilidad básica, estado cambiante de la base de datos y consistencia eventual. Estas bases de datos están pensadas para almacenar grandes cantidades de información, esto implica que para su adecuado funcionamiento consuman más recursos de hardware, por tal motivo, se piensa entonces en un sistema distribuido en el que muchos nodos soporten en simultaneo la funcionalidad de la base de datos. Sin embargo, para un sistema distribuido resulta muy difícil garantizar la consistencia de la información, la tolerancia a particiones y la disponibilidad todas en simultaneo. Este principio es conocido como el teorema CAP(del Busto & Enr'\iquez, 2012) por sus sígalas en ingles Consistency, Aviability and Partition Tolerance, este principio plantea que en los sistemas distribuidos solo se puede tener dos de las tres características en simultaneo, por lo tanto es importante elegir dependiendo la necesidad, la alternativa de base de datos que más se ajuste a la solución. En ese orden de ideas, sí lo más importante es la consistencia de la información, será

preciso optar por una base de datos relacional, este es el caso de las implementaciones en bancos, en las cuales lo más importante es la integridad de los datos por encima de la disponibilidad. Sin embargo, si lo que se busca es tener una alta disponibilidad, con tiempos de respuesta muy bajos, entonces será preciso optar por las bases de datos no relacionales. Cabe resaltar que las bases de datos NoSQL ofrecen una alternativa para las bases de datos relacionales, no son un remplazo.

Resulta oportuno aclarar que las bases de datos no relacionales se pueden clasificar por el modelo de datos que utilizan, es decir, la forma en la que almacenan la información. En esta taxonomía se encuentran las bases de datos documentales que son aquéllas en las que cada registro corresponde a un documento de cualquier tipo, por ejemplo, revistas, artículos de revista, documentos electrónicos, capítulos de libros, tesis doctoral, entre otros. Estas sirven para almacenar grandes volúmenes de información, generalmente utilizando una estructura simple como JSON o XML y donde se utiliza una clave única para cada registro. Las más utilizadas en esta categoría son “RavenDB”, “TerraStore”, “CouchDB”(Foundation, 2017) y “MongoDB”(MongoDB, s. f.), esta última maneja Json como formato para estructurar los datos y la información. Otra clasificación se da con las bases de datos de tipo Clave-Valor en las cuales se encuentran “Redis”(Redislabs, 2017), “Membase”, “Voldemort”,”MemcachceDB”.

Por otra parte, los métodos de investigación utilizando software para capturar o producir datos digitales con el objetivo de investigar diferentes aspectos de internet se han utilizado durante más de una década. Estos datos pueden explotarse para analizar los fenómenos culturales (costumbres, vestimenta, religión, etc.) que para este caso se presentan en las redes sociales. Los métodos digitales (Haas & Grams, 2000) tienen un número de ventajas en comparación con los tradicionales y se pueden ver reflejados en costo, velocidad, esfuerzo requerido, detalle etc. Cabe resaltar que en medio existen algunas consideraciones éticas a la hora de realizar extracción de datos pero estas se encuentran delimitadas por las políticas de privacidad(Facebook, 2016) de los usuarios de Facebook para el caso particular .

Por otra parte, las aplicaciones permiten que los usuarios puedan limitar o curar la forma en que la información allí publicada es accesible a los demás, esto es importante porque al hacer uso de estas plataformas las personas deben aceptar las políticas de privacidad (Facebook, 2016) y generalmente no existe la posibilidad de que los usuarios modulen los derechos que conceden, la solicitud debe pedir permisos detallados para el uso de elementos individuales, pero el usuario sólo

puede aceptar a todas las solicitudes o simplemente no utilizar la plataforma. El acceso puede ser revocado después de la instalación, pero esto significa que las aplicaciones pueden leer esos datos al menos una vez.

Sin embargo, la extracción de datos adjunta al presente proyecto busca recolectar datos de cuentas públicas de entidades o personajes en un contexto político, esto implica que un alto porcentaje de la información allí contenida sea expuesta como un conjunto de datos públicos y accesible a todas las personas, por tal motivo, los datos pueden ser captados teniendo en cuenta las normativas de tratamiento de datos sin que se vea afectada la privacidad de las personas.

En marzo de 2017 Facebook anunció que había alcanzado alrededor de un billón (Zuckerberg, Mark, Andreessen, 2017) de usuarios activos en red, esto posiblemente lo convierte en uno de los mayores medios de comunicación de la historia de la humanidad, por tal motivo no es extraño ver que investigadores de muchas áreas de la ciencia y de la sociedad se han movido rápidamente para utilizar la plataforma para realizar sus estudios. Aunque los métodos tradicionales como entrevistas, experimentos y observaciones aún siguen siendo altamente utilizados, existen otros estudios basados en “Data Crawling”(Thelwall, 2001) o “Rastreo de Datos”, es decir, recopilación de información sobre usuarios de sus perfiles sin su participación activa (Wilson, Gosling, & Graham, 2012) que permiten realizar analítica de datos de manera más precisa.

Existen diferentes técnicas para realizar extracción de datos web. Un sistema de extracción de datos web según (Baumgartner, Gatterbauer, & Gottlob, 2009) es un software que extrae, de forma automática y repetida, datos de páginas web con contenido cambiante y que entrega datos extraídos a una base de datos o alguna otra aplicación. HTML es el lenguaje que predomina en la implementación de páginas web y está apoyado por W3C (World Wide Web Consortium) que es un consorcio internacional que genera recomendaciones y estándares para el desarrollo de Internet a largo plazo. Las páginas HTML pueden considerarse como una forma de datos, cabe resaltar que existe grandes cantidades de información en formatos que no son HTML, como lo son mensajes de correo electrónico, código de software y documentación relacionada a este, etc. Una de las características más explotadas en la extracción de datos es la naturaleza semiestructurada de las páginas web, estas pueden ser representadas como arboles ordenados, las páginas web HTML están conformadas básicamente por texto sin formato que contiene etiquetas HTML, palabras clave específicas definidas en el lenguaje de marcado,

que el navegador interpreta para representar los elementos específicos de la página, por ejemplo, links, botones, imágenes entre otros.

Las etiquetas HTML pueden anidarse una en otra, formando una estructura jerárquica, que puede tratarse de forma más organizada para hacer la extracción. Los procedimientos que buscan extraer datos de fuentes no estructuradas o semiestructuradas como las páginas web son llamados “Web Wrapper”, según (Ferrara, De Meo, Fiumara, & Baumgartner, 2014) es un procedimiento, que podría implementar uno o varios algoritmos, que busca y encuentra los datos requeridos por un usuario humano, extrayéndolos de fuentes Web no estructuradas (o semi-estructuradas) y transformándolos en datos estructurados, fusionando y unificando esta información para su posterior tratamiento, de forma semiautomática o totalmente automática.

Los Web Wrappers tienen un ciclo de vida constituido principalmente por tres fases(Ferrara et al., 2014):

- ✓ Generación del wrapper, se define de acuerdo con algunas técnicas.
- ✓ Ejecución del wrapper, el wrapper se ejecuta y extrae datos continuamente.
- ✓ Mantenimiento del wrapper, la estructura de las fuentes de datos (páginas web) puede hacer cambiar el wrapper.

Adicionalmente existen tres aspectos importantes que describen de forma general el proceso de extracción de datos web.

- ✓ **Automatización y Extracción**

La automatización del acceso a las páginas web, así como la localización de sus elementos es una de las características más importantes en los sistemas de Extracción de Datos Web.

- ✓ **Transformación de datos**

Los pasos entre la extracción y la entrega son denominados como transformación de datos, los datos pasan por fases de limpieza y resolución de conflictos, una vez completadas estas fases los usuarios alcanzan el objetivo para obtener información limpia y estructurada.

- ✓ **Uso de los datos extraídos**

Los datos adquiridos se estructuran en el formato necesario, estos datos están listos para ser utilizados, el último paso es entregarlos a un sistema de gestión que como lo nombramos anteriormente se puede tratar de una base de datos relacional o no relacional. Estos datos pueden utilizarse genéricamente con fines analíticos o estadísticos o simplemente volver a publicarlos en un formato estructurado.

Para el presente proyecto las técnicas de extracción están dadas de forma implícita por el API que proporciona la comunidad de desarrolladores de Facebook llamada Graph API, esta es la principal vía para introducir datos en la plataforma de Facebook y extraer información. Es una API de bajo nivel basada en HTTP que permite consultar datos, publicar historias nuevas, administrar anuncios, subir fotos y realizar varias tareas adicionales mediante programación que puede implementar una aplicación(Facebook, 2017).

Teniendo en cuenta todos los conceptos anteriormente mencionados, la aplicación de técnicas enfocadas en minería web en redes sociales, almacenamiento adecuado utilizando bases de datos no relacionales debido a las condiciones de grandes volúmenes de datos y los requerimientos del sistema, todos integrados en un proceso de desarrollo de software definido por una metodología ágil, permitirán la construcción de un componente efectivo para la extracción y almacenamiento de datos de redes sociales.

# 4. MARCO CONCEPTUAL

## 4.1 DATA RETRIEVAL

La recuperación de datos o Data Retrieval es el proceso mediante el cual se identifican y se extraen datos de una base de datos por medio de una consulta proporcionada por el usuario o la aplicación (Aho & Ullman, 1979). Permitiendo la obtención de información de una base de datos con el fin de mostrarla en un monitor y / o utilizarla dentro de una aplicación.

La recuperación de datos normalmente requiere escribir y ejecutar comandos de recuperación o extracción de datos o consultas en una base de datos. Con base en la consulta proporcionada, la base de datos busca y recupera los datos solicitados.

Las aplicaciones y el software generalmente utilizan varias consultas para recuperar datos en diferentes formatos. Además de datos simples o más pequeños, la recuperación de datos o Data Retrieval también puede incluir la recuperación de grandes cantidades de datos, generalmente en forma de informes.

## 4.2 WEB SCRAPING

Web Scraping se refiere a métodos utilizados para recopilar información a través de Internet. Generalmente por medio de software que simula la navegación de un humano para recoger pedazos especificados de la información de diversos sitios web (Glez-Peña, Lourenço, López-Fernández, Reboiro-Jato, & Fdez-Riverola, 2014). Aquellos que usan programas de web scraping pueden buscar y recoger ciertos datos para vender a otros usuarios, o para usarlos con fines promocionales en un sitio web. El raspado de Web también se llama extracción de datos Web, recolección de Web o raspado de pantalla.
En esencia web scraping es una forma de minería de datos. Artículos como informes meteorológicos, precios de mercado, o cualquier otra lista de datos recopilados se pueden buscar en múltiples sitios por medio de web scraping.

La práctica de web scraping puede chocar con las condiciones de uso de algunos sitios web que no permiten ciertos tipos de minería de datos. A pesar de los desafíos legales, el web scraping promete convertirse en una forma popular de recopilar información a medida que estos tipos de recursos de datos agregados se vuelven más poderosos.

## 4.3 WEB MINING

Minería Web es una rama de la minería de datos que se concentra en internet como la fuente de datos principal, incluyendo todos sus componentes del contenido web (Kosala & Blockeel, 2000). El contenido de los datos extraídos de la Web pueden ser una colección de hechos que las páginas web contienen, y pueden consistir en texto, datos estructurados como listas y tablas e incluso imágenes, vídeo y audio. La minería web es el proceso de utilizar técnicas y algoritmos de minería de datos para extraer información directamente de la web extrayéndola de documentos y servicios web, contenido web, hipervínculos y registros de servidor. El objetivo de la minería web es buscar patrones en los datos web mediante la recopilación y el análisis de información con el fin de obtener una visión de las tendencias, la industria y los usuarios en general.

En este proceso se extrae la información útil a partir del contenido de páginas web y documentos web, que son en su mayoría textos, imágenes y archivos de audio y vídeo. Las técnicas utilizadas en esta disciplina han sido fuertemente extraídas del procesamiento del lenguaje natural y la recuperación de la información.

## 4.4 BIG DATA

Se utiliza cuando las técnicas tradicionales de minería y manipulación de datos no pueden revelar las ideas y el significado de los datos tratados. Los datos no estructurados o sensibles al tiempo o simplemente muy grandes no pueden ser procesados por motores de base de datos relacional. Este tipo de datos requiere un enfoque de procesamiento diferente denominado Big Data (McAfee, Brynjolfsson, Davenport, Patil, & Barton, 2012), que utiliza paralelismo masivo en hardware fácilmente disponible. En pocas palabras, Big data refleja el mundo cambiante actual. Cuanto más cambian las cosas, más se capturan y se registran como datos. Procesar la información de la siguiente manera muestra por qué Big Data se ha vuelto tan importante:

- ✓ La mayoría de los datos recopilados ahora no están estructurados y requieren diferentes procesos de almacenamiento y procesamiento que los que se encuentran en las bases de datos relacionales tradicionales.

- ✓ El poder computacional disponible es cada vez mayor, lo que significa que hay más oportunidades para procesar grandes cantidades de datos.

- ✓ Internet ha democratizado los datos, aumentando constantemente los datos disponibles y produciendo cada vez más datos en bruto.

## 4.5 RESTFUL WEB SERVICES

Los servicios web son aplicaciones modulares que exponen la lógica del negocio como servicios a través de Internet por medio de interfaces programables y protocolos como tpc/ip se pueden usar para encontrar la forma de invocar esos servicios(Wagh & Thool, 2012). En otras palabras, los servicios web son una tecnología que mediante el uso de estándares y protocolos de comunicación permite el intercambio de información entre aplicaciones. Esto es muy importante porque permite que aplicaciones se comuniquen entre si, sin importar el lenguaje en el que estén implementadas, esto es porque la información viaja en formatos de texto definidos que pueden ser extensible markup lenguaje o bien JavaScript Object Notation, entre otros. Para el desarrollo del presente proyecto se utiliza el formato Json. Existen diferentes arquitecturas que se emplean en los servicios web, entre ellos está la de Transferencia de Estado Representacional, está permite hacer llamados y peticiones mediante el protocolo HTTP utilizando los métodos de acceso GET, POST, PUT, DELETE(Fielding et al., 1999). Por otra parte, el uso de servicios web basados en la arquitectura restful brinda grandes ventajas sobre los servicios web tradicionales soap. Algunas de ellas son(Wagh & Thool, 2012):

- ✓ Consume menos ancho de banda porque su respuesta es liviana.
- ✓ Los restful web services se pueden consumir con simples solicitudes GET, los servidores proxy intermedios pueden almacenar en caché su respuesta con mucha facilidad.
- ✓ Los servicios web restful proporcionan flexibilidad con respecto al tipo de datos devueltos, mientras que los servicios web soap siempre devuelven datos XML.
- ✓ El cambio de servicios web restful no requiere ningún cambio en el código del lado del cliente.

# 5. METODOLOGÍA

La construcción de la metodología a utilizar en el proyecto está enmarcada por varias características importantes, teniendo en cuenta que el desarrollo del proyecto está dado por una sola persona y que se cuenta con limitaciones de tiempo para el desarrollo del proyecto de acuerdo a su alcance, se investigó sobre la metodología que más se adaptara a las necesidades del proyecto, tras evaluar diferentes alternativas y metodologías posibles se optó porque el proceso de desarrollo del "Componente de extracción y almacenamiento" estaría basado en la metodología de Programación Extrema para solistas (Casallas, 2012), esta es una metodología ágil de desarrollo de software. Las características de esta metodología permiten realizar una programación más organizada y simple teniendo en cuenta la definición de los objetivos específicos, y la identificación de los requisitos del sistema.

Una vez realizado el diseño del componente, se realiza la tercera etapa que consiste en la codificación de dicho componente. En este punto existen grandes ventajas utilizando la metodología de programación extrema ya que esta busca que el código sea sencillo y entendible, permitiendo un mejor ambiente de trabajo por parte del programador. A esta codificación se le aplicaran pruebas de aceptación, esta será la cuarta etapa y tendrá como objetivo encontrar posibles errores o falencias para corregirlos, esto se puede conseguir haciendo organizado el software en versiones, que van corrigiendo los errores encontrados, hasta lograr el propósito deseado.

La programación extrema es una metodología ágil para desarrollo de software basada en cuatro principios que son simplicidad, comunicación, feedback o retroalimentación y coraje. Esta metodología está diseñada para ser usada en pequeños grupos de desarrollo, en el caso particular para una sola persona.

Las prácticas de programación extrema (Lappo, 2005) para solistas son:

- ✓ **Small Releases**

Poner en producción un sistema simple desde el comienzo y actualizarlo frecuentemente en ciclos muy cortos.

- ✓ **Design Metaphor**

Usar un sistema de nombres y una descripción común. Vocabulario común

- ✓ **Simple Design**

Un programa construido con programación extrema debe ser el más simple que satisfaga los requerimientos. No se desarrolla “para el futuro”. Se hace énfasis en lo que tiene valor para el cliente.

- ✓ **Testing**

Clientes proveen pruebas de aceptación para asegurarse que los aspectos que ellos requieren son provistos por el software.

- ✓ **Refactoring**

Programación extrema mejora constantemente el diseño del sistema haciendo refactoring. Esfuerzo por mantener el sistema sin código duplicado, simple, cohesivo, etc.

- ✓ **Coding Standards**

El programador debe escribir y documentar el código en la misma manera.

Ventajas de Programación Extrema

- ✓ Código valioso y más organizado en producción anticipada.
- ✓ Más fácil para el cliente cambiar su mente.
- ✓ Robusto conjunto de pruebas para todo el ciclo de vida.
- ✓ Vista precisa del estado del proyecto.
- ✓ Cierre de funciones y menor tasa de errores.

Desventajas de Programación Extrema

- ✓ Es recomendable emplearla solo en proyectos a corto plazo.
- ✓ Altas comisiones en caso de fallar.
- ✓ Puede no siempre ser más fácil que el desarrollo tradicional.

# 6. DESARROLLO DEL PROYECTO

En esta sección del documento se presenta el proceso de desarrollo llevado a cabo, teniendo en cuenta las técnicas, métodos y herramientas utilizadas para la realización del proyecto, tomando como punto de partida la fase de investigación, posteriormente las fases de análisis, diseño e implementación y la fase final de pruebas y resultados.

## 6.1 INVESTIGACIÓN

La fase de investigación fue la fase inicial del proyecto, esta fase fue indispensable para el desarrollo porque permitió indagar y conocer no solo los conceptos referentes a la recolección de datos, sino también las herramientas y técnicas de extracción que podrían ser de utilidad a la hora del diseño e implementación del proyecto. Dentro de la fase de investigación se tomaron algunos temas de referencia sobre los cuales se centró la investigación, estos temas comprendían la recuperación de información o data retrieval, almacenamiento de la información y posibles tecnologías a utilizar en el proyecto.

Para realizar una adecuada fase de investigación se tomaron fuentes de información provenientes de bibliotecas virtuales, bases de datos científicas, papers académicos y diferentes páginas oficiales y recursos web concernientes a la investigación. Por otra parte, se realizaron filtros de la información pertinente por medio de palabras clave referentes al contexto, de igual manera se procuró tener en cuenta bibliografía y referencias filtradas por año de publicación, buscando en lo posible acceder a la información más actual, sobre todo en el aspecto tecnológico y herramientas desarrolladas.

Para lograr plantear y desarrollar el componente de extracción y almacenamiento adjunto al presente proyecto, se realizó un proceso extenso de investigación comprendido en un periodo de tiempo de más o menos ocho meses, teniendo en cuenta el tiempo empleado en trabajo de grado y en la fase previa de formulación. En este proceso se vinculan información catalogada como verídica por que proviene de las fuentes anteriormente señaladas.

Con referencia a lo anterior, cabe resaltar que, si bien la fase de formulación necesita de un proceso constante de investigación, las fases siguientes del proyecto también lo hacen, indicando que este no es un proceso que se cierre definitivamente, con esto se hace referencia a que el proceso de investigación es el punto de partida como fase inicial, pero además está presente en las demás etapas y transcurrir del proyecto.

## 6.2 ANÁLISIS Y PLANIFICACIÓN

Es la fase continua a la fase de investigación, es donde se determinaron las bases del proyecto, definiendo los pasos a seguir mediante la selección de una metodología para el desarrollo del proyecto.

En la fase de análisis y planificación se debió tener en cuenta además de lo anteriormente mencionado, los procesos y las posibles actividades que se requieren para llevar a cabo el componente de extracción y almacenamiento según la investigación realizada hasta esta fase.

Es esencial resaltar la importancia de la fase de análisis y planificación porque en ella se realizó el proceso de levantamiento de requerimientos del componente, este paso es fundamental porque permitió conocer las funcionalidades y las especificaciones que eran necesarias para el desarrollo e implementación de la solución. Sin embargo, previo al levantamiento de requerimientos fue necesario definir la metodología a seguir, la cual es programación extrema para solistas y se encuentra especificada en el inciso de metodología en el presente documento.

En referencia a lo anterior, una vez seleccionada la metodología se realizó el proceso de levantamiento y definición de requerimientos, estos están consignados en el Documento de Especificación de Requerimientos de Software(Anexo A), la realización de este documento es importante por dos razones principales, la primera es que es el punto de partida de los diseñadores y programadores del componente para llevar a cabo las etapas de diseño e implementación del componente; la segunda es porque hace parte fundamental de la documentación del proyecto.

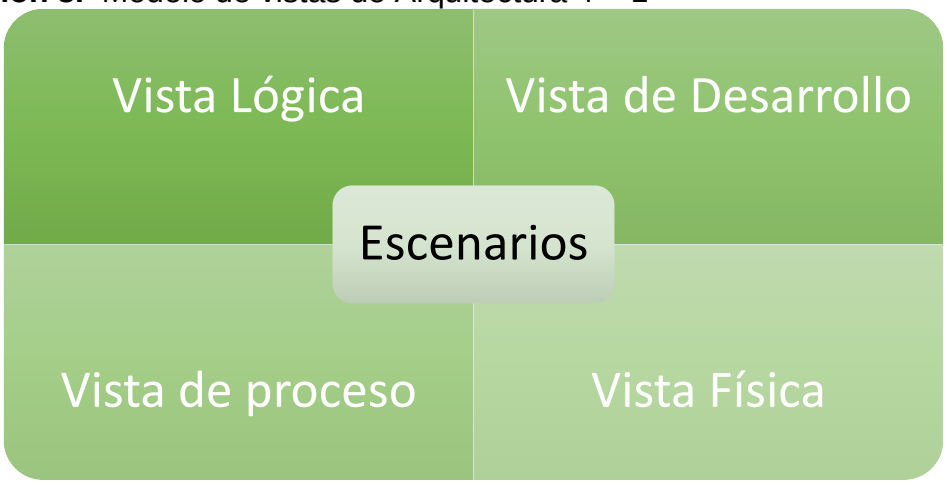
En la fase de análisis y planificación surgen preguntas provenientes del proceso de levantamiento de requerimientos, estas preguntas son acerca de temas referentes a los lenguajes de programación que más se adecuan a la implementación de la solución, la arquitectura de software más pertinente a utilizar, que tipo de almacenamiento utilizar y cómo hacerlo, entre otras dudas que surgen en esta fase, en términos generales se puede decir que surgen dudas de carácter tecnológico, sin embargo este trabajo es propio de las fases de diseño e implementación, fases siguientes del proyecto.

## 6.3 DISEÑO

La fase de diseño del componente de extracción y almacenamiento tenía como objetivo proporcionar una idea clara y completa de lo que es el software, partiendo del levantamiento de requerimientos realizado en la fase de análisis, usualmente el diseño de software se hace con el apoyo de diagramas que están estandarizados mediante el Lenguaje Unificado de modelado UML(Tabares, Pineda, & Barrera, 2008), el principal aporte de los diagramas es tener una perspectiva del software desde diferentes puntos de vista y de esta manera lograr comprender el sistema a desarrollar.

En la fase de diseño se tuvo en cuenta el diseño general del componente desde diferentes puntos, como la arquitectura del sistema, el diseño de los componentes, las interfaces de intercambio de información y datos, entre otras. Particularmente para el diseño de la arquitectura se utilizó el Modelo de vistas de Arquitectura 4+1(Kruchten, 1995), este modelo pretende dar una descripción de la arquitectura utilizando diferentes vistas de concurrencia (Ver Ilustración 3), estas vistas son la vista lógica, que se encarga de modelar la funcionalidad del sistema, la vista de proceso, que se encarga de describir la escalabilidad o rendimiento del sistema, la vista de desarrollo, esta vista está enfocada en los programadores de la solución, y finalmente la vista física, que se encarga de describir al sistema en hardware y comunicaciones. En referencia a lo anterior, la explicación en detalle y las vistas mencionadas se encuentran en el Documento de Arquitectura de Software (Anexo C).

**Ilustración 3.** Modelo de vistas de Arquitectura 4 + 1




Fuente: Elaboración propia.

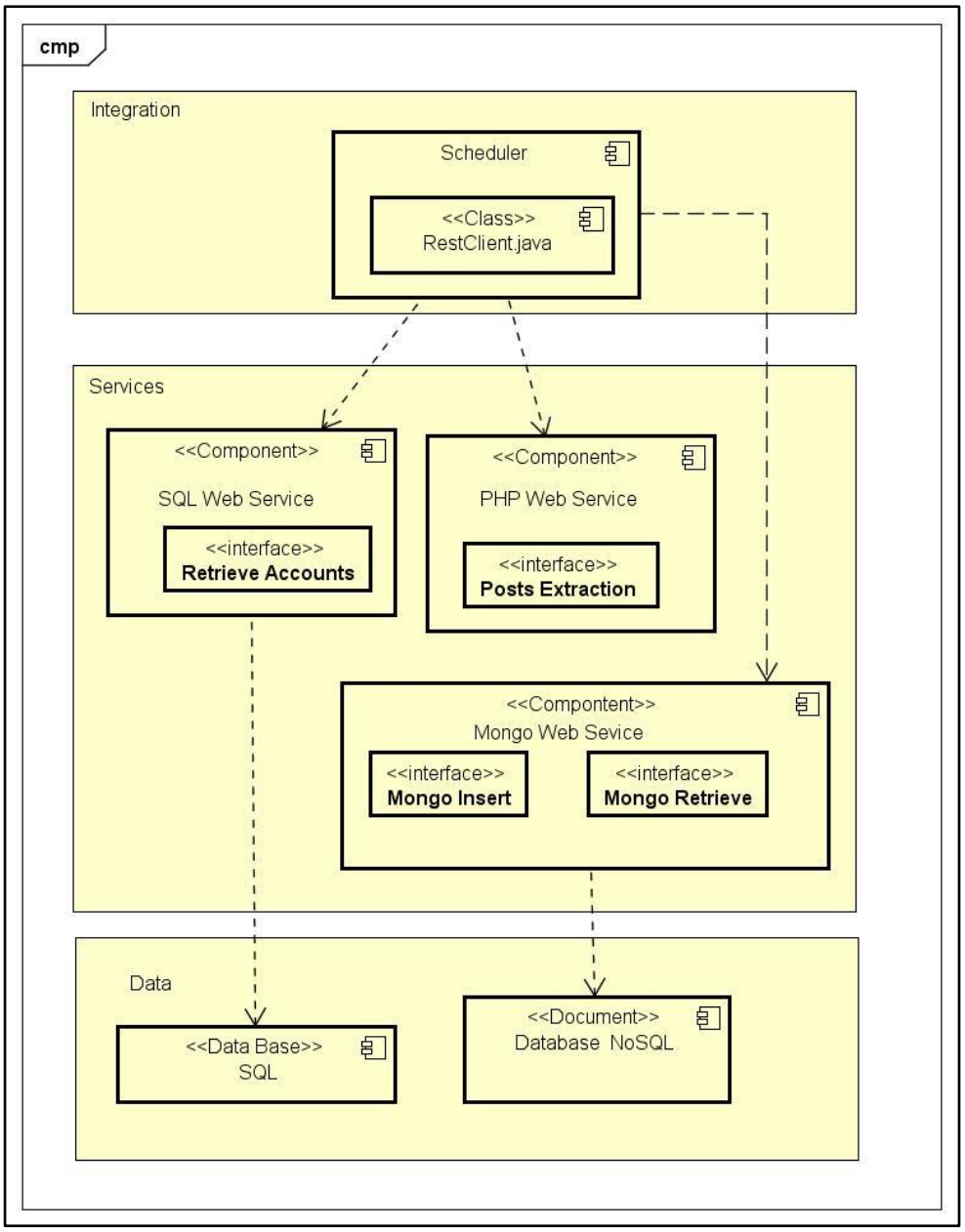
Por otra parte, se tuvieron en cuenta las restricciones, suposiciones y dependencias para el desarrollo del componente de extracción y almacenamiento, estas especifican las restricciones generales tanto de software como hardware, de igual manera teniendo en cuenta a las partes involucradas en el desarrollo del proyecto.

Las restricciones, suposiciones y dependencias están descritas en detalle en el Documento de Diseño de Software (Anexo B).

A continuación, se exponen y describen los diagramas de componentes y de despliegue del componente de extracción y almacenamiento:

### 6.3.1 Diagrama de componentes

**Ilustración 4.** Diagrama de componentes.




Fuente: Elaboración propia.

*Scheduler:* Este componente es el encargado de integrar toda la funcionalidad del "Componente de extracción y almacenamiento", interactuando con los servicios web, permitiendo la ejecución de manera ordenada de los diferentes componentes que hacen parte del sistema.

*SQL Web Service:* Es el componente encargado de consultar las llaves de autenticación y las cuentas sobre las cuales se quiere extraer información, esta información será utilizada respectivamente como las entradas del componente de extracción php.

*PHP Web Service:* Además de comunicarse con el Scheduler, este componente puede considerarse como el corazón del sistema debido a su importancia, es el encargado de la extracción de las publicaciones o posts de la red social Facebook. Por medio del API "Graph Api" y de las credenciales de acceso obtenidas por el componente SQL extrae los datos de las cuentas requeridas en formato Json y los retorna para que puedan ser almacenados respectivamente.

*Mongo Web Service:* Se encarga de gestionar el almacenamiento de la información entrante por parte del componente de extracción Php y consultar la información contenida en la base de datos NoSQL, por medio de dos interfaces llamadas Mongo Insert y Mongo Retrieve, ambas interfaces intercambian información en formato Json.

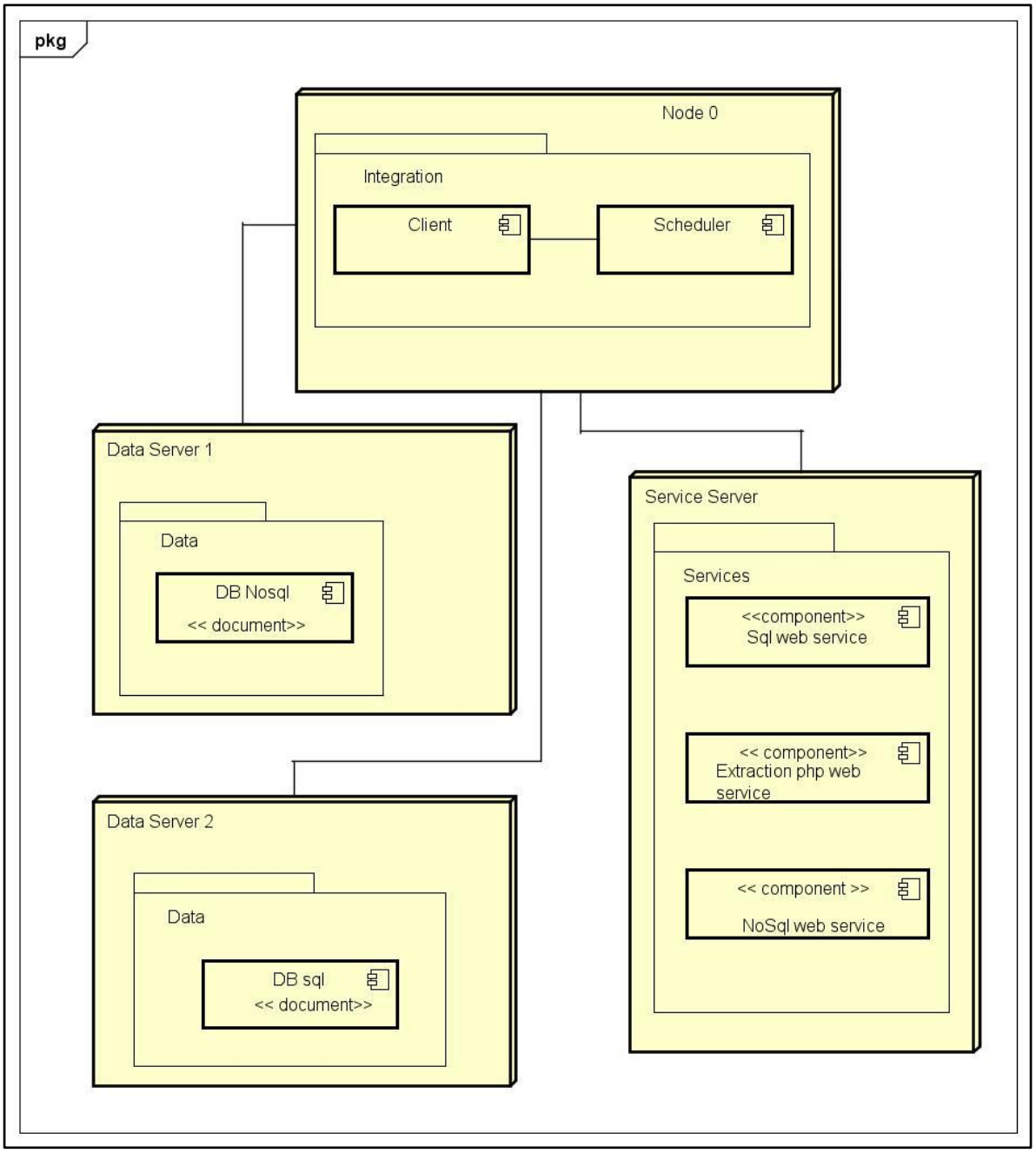
*SQL Database:* Este componente se encuentra en la capa de datos de la arquitectura del componente general, se encarga de almacenar las llaves de acceso al API de extracción, de igual manera, contiene las cuentas de las páginas de las que se extraerán los posts o publicaciones.

*NoSQL Database:* Este componente al igual que el anterior, se encuentra en la capa de datos, es el encargado de almacenar toda la información extraída por parte del componente Php, es NoSQL por la ventaja en el almacenamiento de grandes cantidades de información en comparación a una base de datos relacional.

Por otra parte, el diagrama de componentes es uno de los diagramas que describe de desarrollo del proyecto, según el modelo de vistas citado anteriormente, los demás diagramas, así como las vistas respectivas se encuentran descritos en detalle en el documento de arquitectura de software (Anexo C).

### 6.3.2 Diagrama de Despliegue

**Ilustración 5.** Diagrama de despliegue.




Fuente: Elaboración propia.

El diagrama de despliegue permite ilustrar cómo funciona el sistema físicamente, la descripción de las características óptimas para cada máquina o servidor para realizar la instalación del sistema se encuentra en detalle en el inciso *5.3 Vista Física*

del Anexo C. En el diagrama de despliegue se ven los componentes físicos del sistema, juntos representan la arquitectura física del componente de extracción y almacenamiento. La arquitectura física está compuesta por cuatro máquinas principales llamadas Nodo0, Data Server 1 y 2, Service Server. Cabe resaltar que, aunque en se tienen cuatro máquinas, dada la codificación del componente, se podría llegar a escalar en más máquinas, teniendo el módulo de servicios dividido en el que cada servicio podría estar alojado en una maquina diferente.

Para dar por concluido, la fase de diseño jugó un papel fundamental en el desarrollo del proyecto porque en esta etapa se dio claridad del sistema al cual se quería llegar en la fase de implementación, la fase de diseño se encuentra en detalle en los anexos del presente documento.

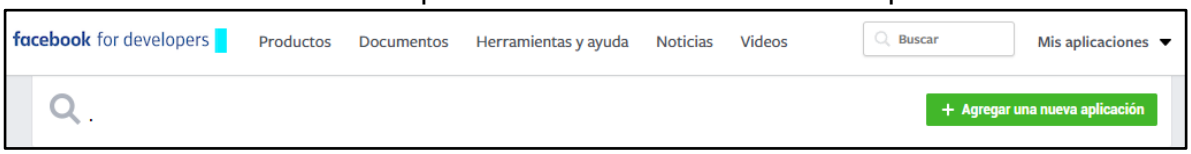
## 6.4 IMPLEMENTACIÓN

La etapa de implementación es el resultado de tener claras las tecnologías a utilizar y de que manera se desarrollara el código para cumplir no solo con los requerimientos del sistema, sino también el diseño propuesto en las fases anteriores. Las especificaciones técnicas concernientes a los lenguajes de programación y ambientes de desarrollo se encuentran en el inciso de restricciones de software del Anexo B. Para la codificación del componente de extracción y almacenamiento se utilizaron los lenguajes orientados a objetos java y php. Haciendo énfasis en la codificación del componente, se realizó un proceso de versionamiento de software, esto permitió tener un mayor control y organización optima sobre el desarrollo gradual del proyecto.

Uno de los retos más grandes a los que se enfrentó el desarrollo del componente fue el extraer los datos de la red social Facebook. Gracias a la fase de investigación tecnológica se indago sobre varias herramientas que permitirían realizar el proceso de extracción, entre las cuales se encontró la interfaz de programación de aplicaciones API llamada “Graph API”, que permitía integrar la extracción de datos de la red social mediante una serie de pasos definidos. Graph Api es una herramienta que provee la comunidad de desarrolladores de Facebook “Facebook for developers” de forma gratuita.

El primer paso para hacer uso de la herramienta era crear una aplicación en la plataforma de Facebook for developers, como se muestra a continuación.

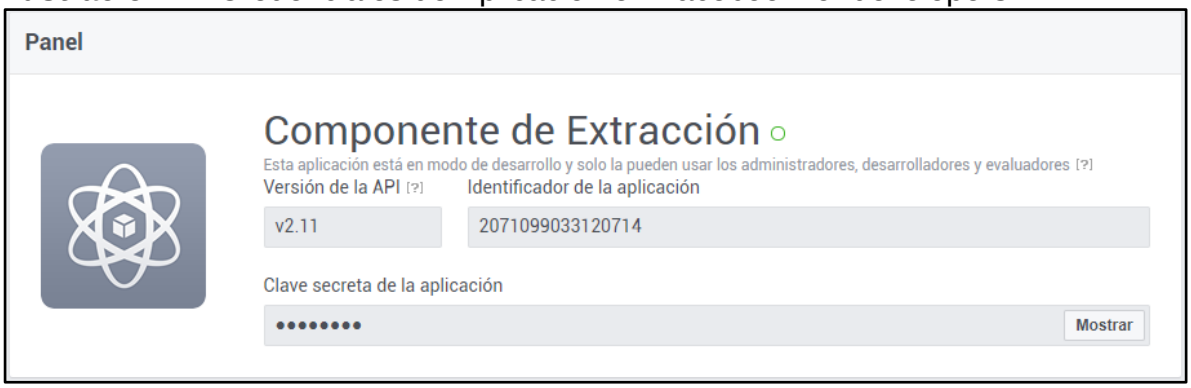
**Ilustración 6.** Creación de Aplicación en Facebook for developers.




Fuente: *https://developers.facebook.com/.*

Una vez creada la aplicación, ésta genera por defecto credenciales de acceso en los cuales se define la versión del API (Ver Ilustración 7), un identificador de aplicación y una clave para poder hacer uso de la aplicación. Cabe resaltar que las credenciales de acceso son fundamentales en el componente de extracción, porque son los que permiten hacer la autenticación a la hora de realizar la extracción desde el código escrito en php. El componente de extracción está escrito en el lenguaje php, esto debido a que el API soporta la integración con este lenguaje, por otra parte, el componente de almacenamiento está desarrollado en Java, y desde este lenguaje se hace el llamado al componente de extracción por medio de la capa de integración.

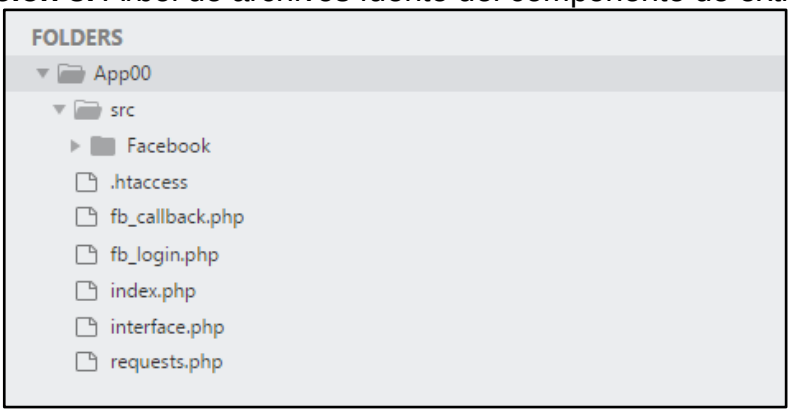
**Ilustración 7.** Credenciales de Aplicación en Facebook for developers.




Fuente: *https://developers.facebook.com/.*

El componente de extracción incorpora una serie de archivos incluyendo el driver para php proporcionado directamente desde Facebook for developers, mediante el cual es posible hacer uso de Graph API el cual hace una serie de llamadas a las credenciales de acceso, posteriormente genera tokens de autenticación y gracias a ellos es posible obtener las publicaciones o posts de las cuentas requeridas. En la ilustración 8 se ve el árbol de archivos correspondiente al componente de extracción de la herramienta.

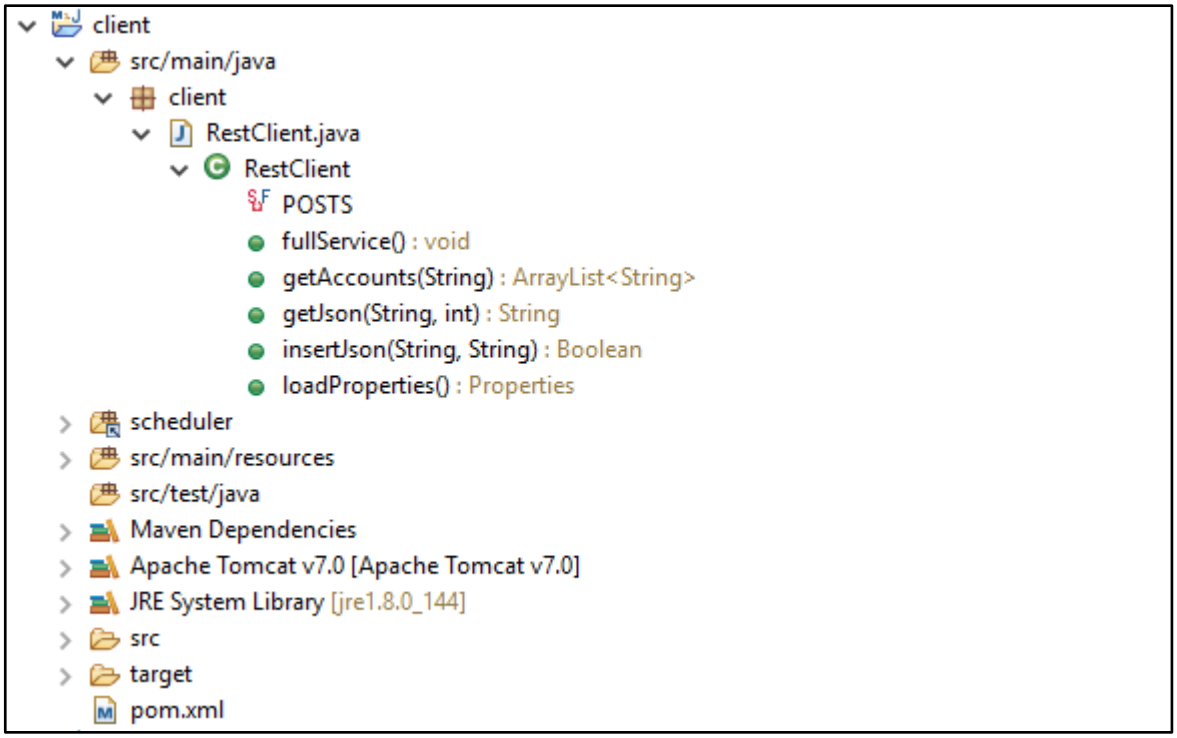
**Ilustración 8.** Árbol de archivos fuente del componente de extracción.




Fuente: Elaboración propia.

Por otra parte, el componente de extracción y almacenamiento intercambia información entre los componentes definidos en el diagrama de componentes (Ver Anexo B) por medio de llamados y consumo de servicios web restful, estos están implementados usando la tecnología JAX-RS(Wagh & Thool, 2012), los componentes de los módulos de integración, datos y servicios, excluyendo el de extracción fueron implementados en java, estos se encuentran distribuidos por proyectos tipo maven(An, 2008) en java, en las ilustraciones 9,10,11,12 se encuentra la estructura de los proyectos que en conjunto forman el componente de extracción y almacenamiento, con sus respectivas clases y métodos.

**Ilustración 9.** Árbol de archivos proyecto client.

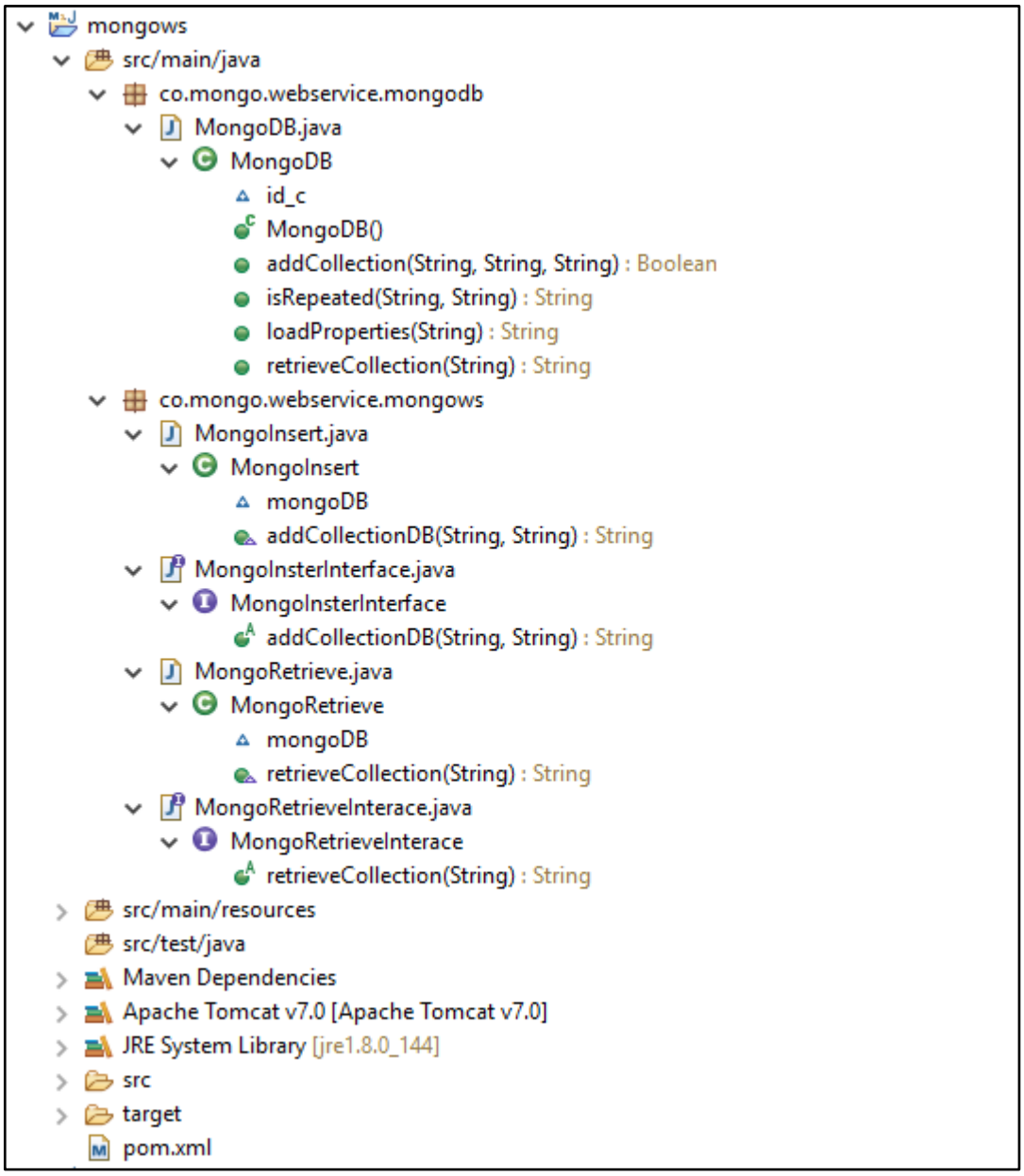



Fuente: Elaboración propia

**Ilustración 10.** Árbol de archivos proyecto scheduler.

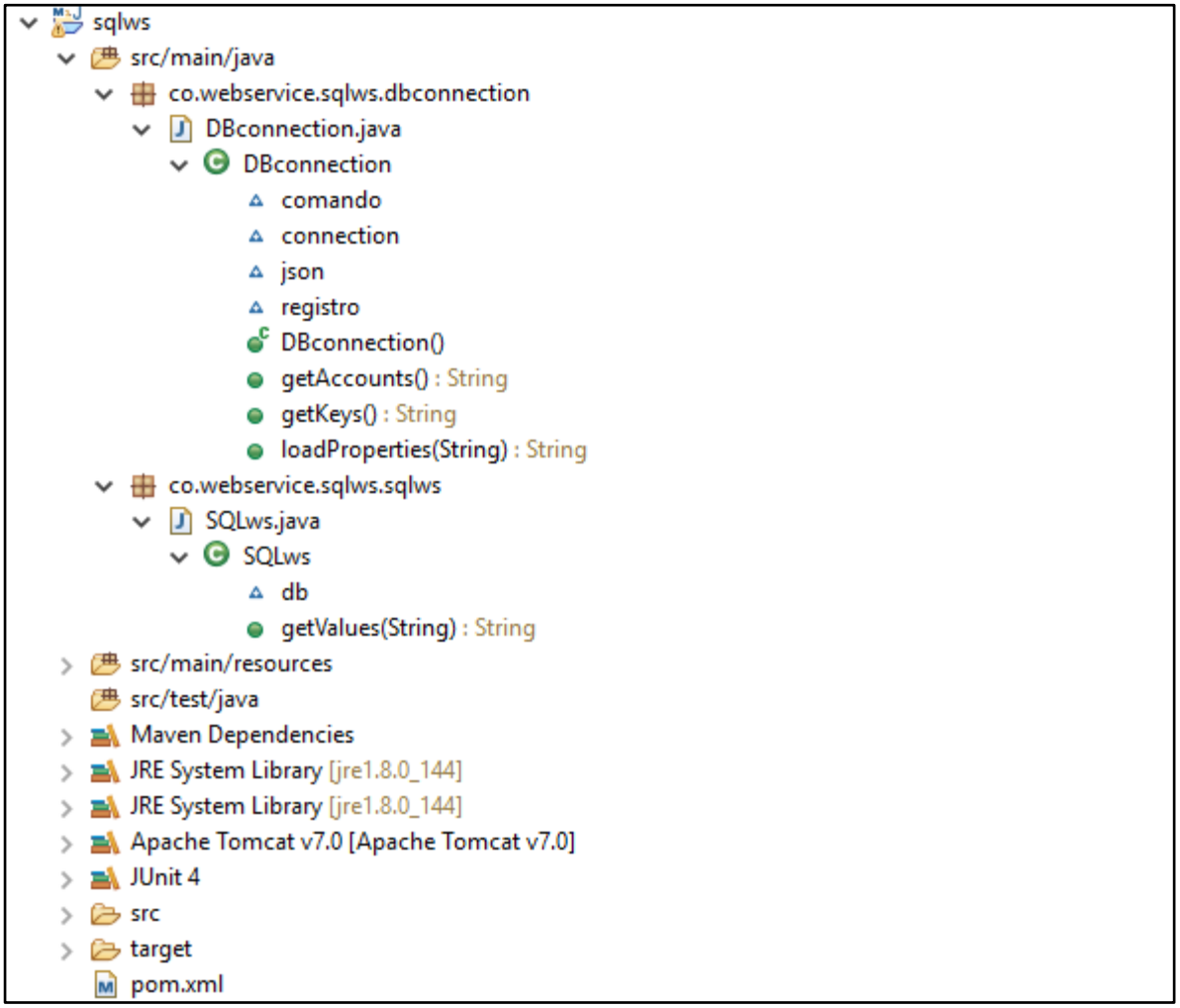

Fuente: Elaboración propia

**Ilustración 11.** Árbol de archivos proyecto mongows.




Fuente: Elaboración propia

**Ilustración 12.** Árbol de archivos proyecto SQLws.




Fuente: Elaboración propia

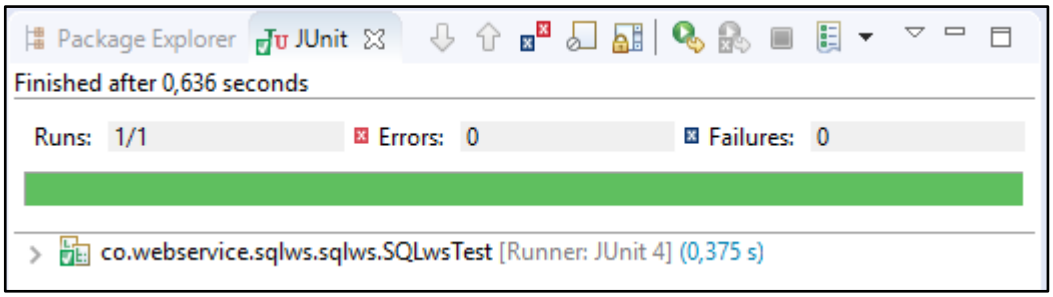
El funcionamiento general del componente esta descrito en el inciso *5.1* del *Anexo C*. Por otra parte, dentro de la fase de implementación se realizó conjuntamente la documentación del código en idioma español, este contempla la definición de cada clase, así como la funcionalidad de cada método teniendo en cuenta los parámetros o atributos de entrada y sus respectivas salidas.

Para concluir, la etapa de implementación de la solución lleva a la última fase dentro del proceso de desarrollo del software que es la fase de pruebas unitarias, cabe resaltar que en esta etapa de llegar a fallar alguna funcionalidad se evalúa y se corrige, de esta manera el ambiente de pruebas y desarrollo van de la mano en el proceso de implementación. De esta manera se da por concluido la fase de implementación dentro del tiempo establecido en la delimitación del proyecto.

## 6.5 PRUEBAS

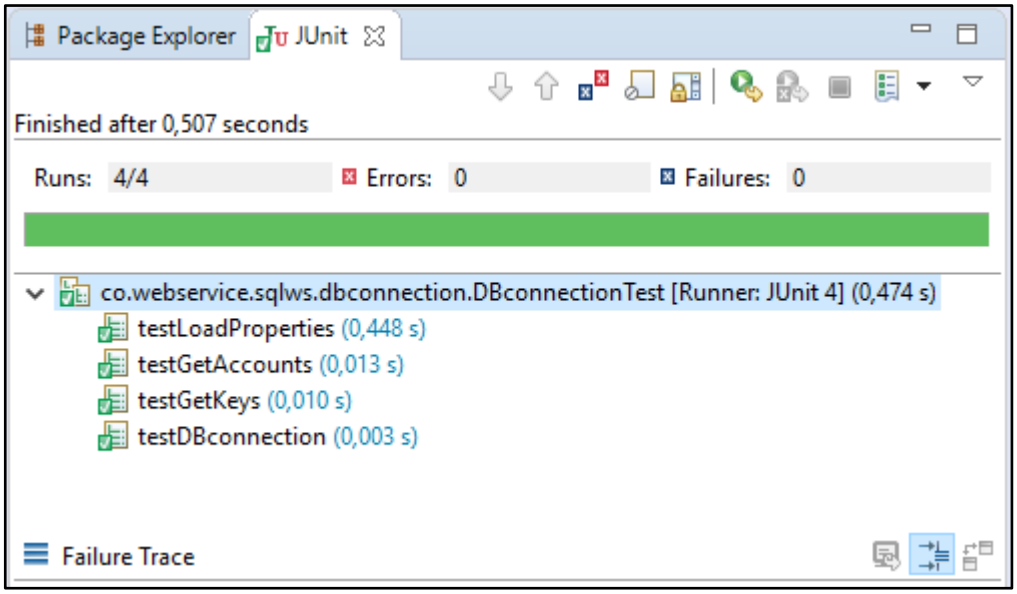
El proceso de pruebas unitarias del componente desarrollado se realizó por medio del software JUnit4 que proporciona un grupo de bibliotecas que permiten evaluar si el comportamiento de cada uno de los métodos de las clases del proyecto se comporta como se espera. Por otra parte, para realizar las pruebas unitarias es necesario programar los test por cada uno de los métodos de cada clase dentro de los proyectos enmarcados en la fase de implementación del componente. A continuación, se presenta el resultado de las pruebas realizadas a las clases y sus respectivos métodos, los resultados destacan el tiempo empleado en la ejecución, así como el número de fallos encontrados.

**Ilustración 13.** Prueba unitaria clase SQLws.




Fuente: Elaboración propia

**Ilustración 14.** Prueba unitaria clase DBConnection.

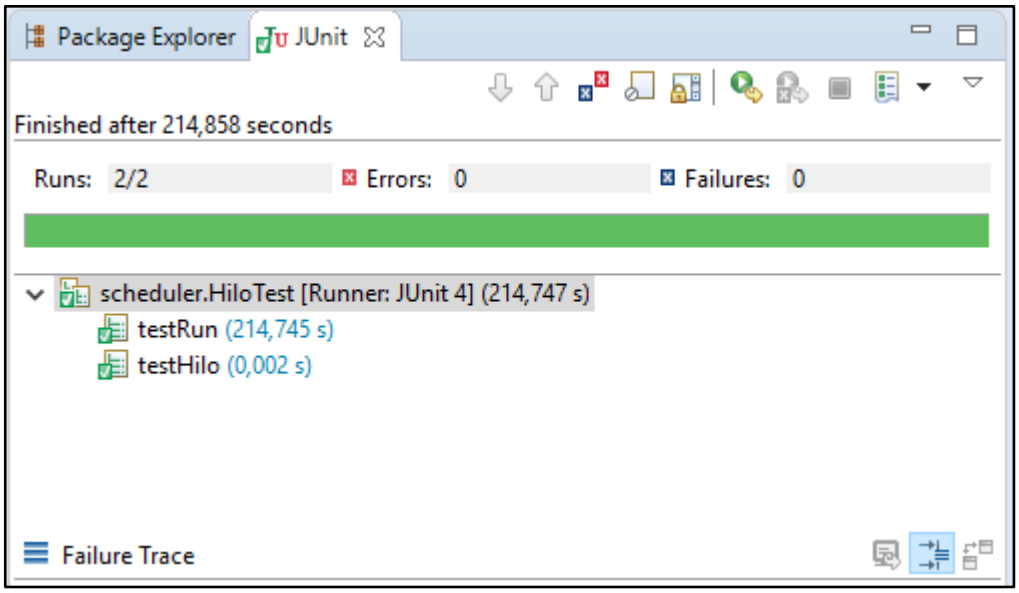



Fuente: Elaboración propia

Las ilustraciones 13 y 14 muestran el resultado de las pruebas unitarias sobre los métodos de las clases SQLws y DBConnection del proyecto SQLws, se evidencia

que los tiempos de respuesta de cada método son bajos, siendo el tiempo más alto de 0,448 s, para el método llamado loadProperties().

**Ilustración 15.** Prueba unitaria clase Hilo.




Fuente: Elaboración propia

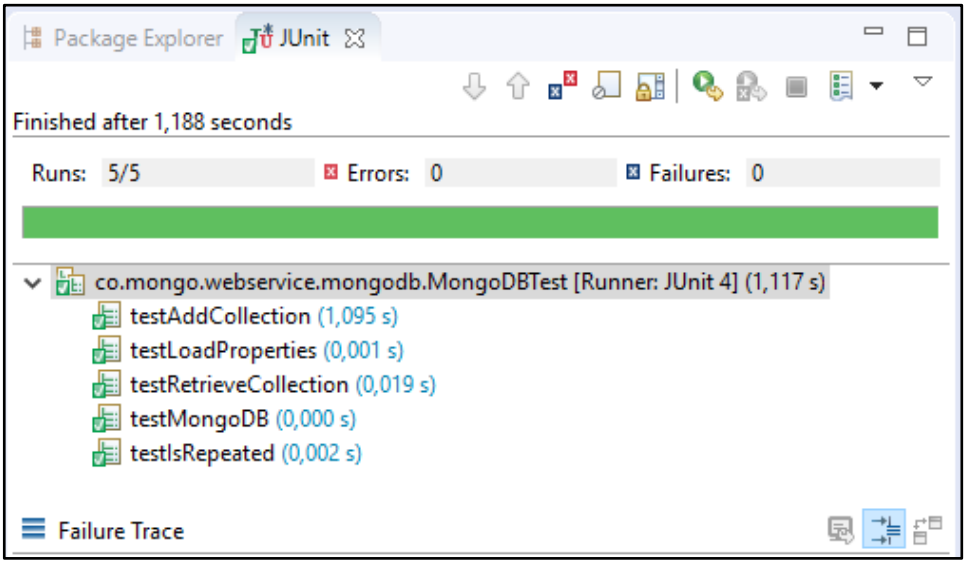
Esta prueba unitaria muestra la ejecución exitosa de la clase Hilo del proyecto Scheduler, el método run() tomó 214,754 segundos en ejecutarse, lo equivalente a más o menos 3 minutos, en este punto hay que resaltar que esta clase es la encargada de la funcionalidad de integración del componente, en la cual se realizo el proceso de extracción y almacenamiento de quince cuentas y por cada una se extrajeron y almacenaron 2000 publicaciones, que en total suman 30.000 publicaciones ver ilustración 16 y 21.

**Ilustración 16.** Cantidad de publicaciones a extraer.

```
29
30  private static final int POSTS = 2000;
```

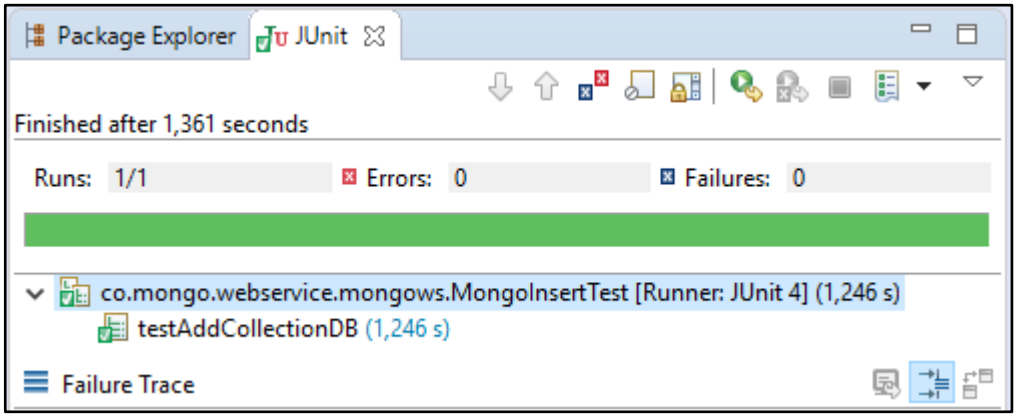
Fuente: Elaboración propia

**Ilustración 17.** Prueba unitaria clase MongoDB.



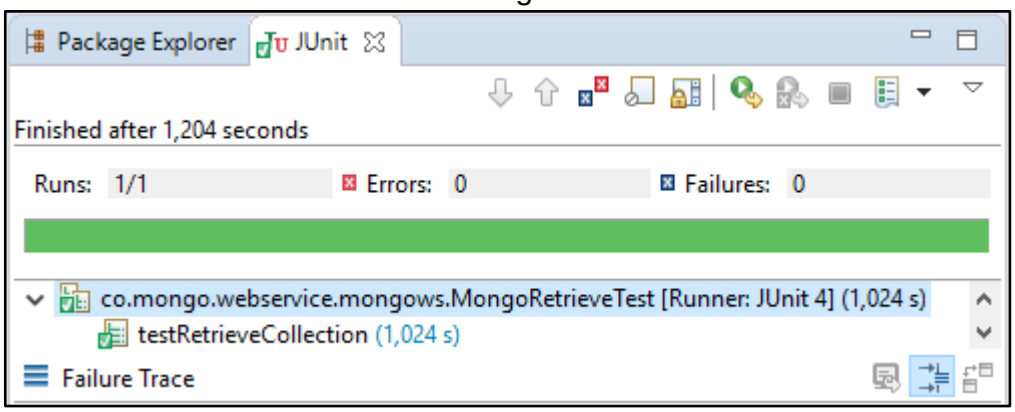

Fuente: Elaboración propia

**Ilustración 18.** Prueba unitaria clase MongoInsert.




Fuente: Elaboración propia

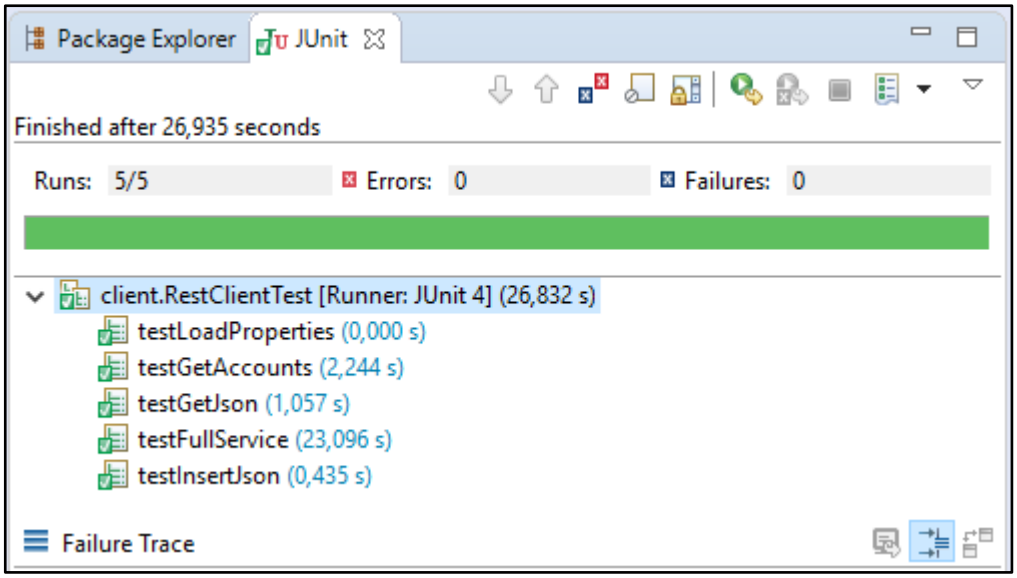
**Ilustración 19.** Prueba unitaria clase MongoRetrieve.




Fuente: Elaboración propia

Las ilustraciones 17,18 y 19 muestran el resultado de las pruebas unitarias sobre los métodos de las clases MongoDB, MongoInsert y MongoRetrieve del proyecto MongoWS, se evidencia que los tiempos de respuesta de cada método son bajos, siendo el tiempo más alto de 1,246 s, para el método llamado addCollection().

**Ilustración 20.** Prueba unitaria clase RestClient.

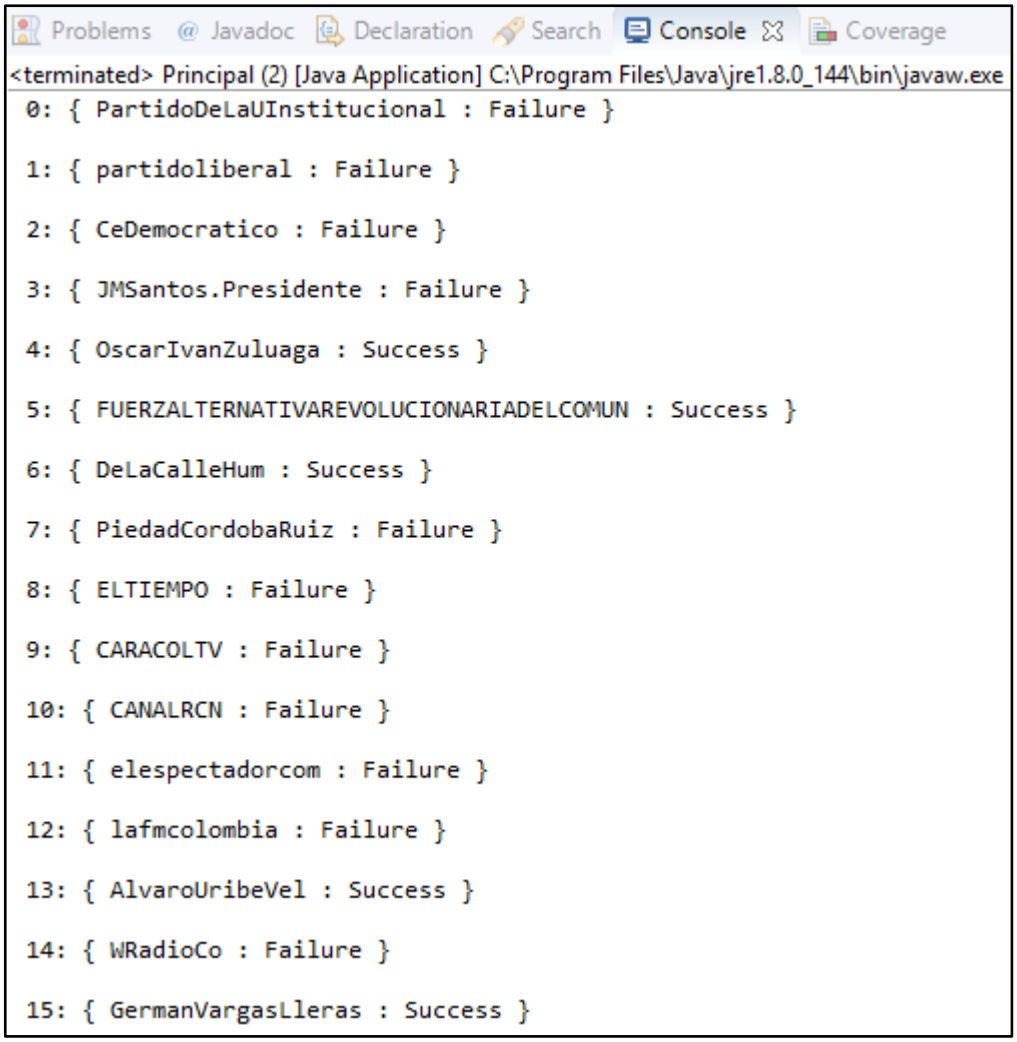



Fuente: Elaboración propia

Las ilustración 20 muestra el resultado de las pruebas unitarias sobre los métodos de las clases RestClient del proyecto client, se evidencia que los tiempos de respuesta de cada método son bajos, siendo el tiempo más alto de 23,096 s, para el método llamado fullService().

Como se puede apreciar en los casos de las pruebas unitarias realizadas a todas las clases del componente de extracción y almacenamiento, estas salieron exitosas, con tiempos de respuesta bajo para el almacenamiento de grandes cantidades de información, sí bien cada método por separado tuvo un resultado positivo en el que se puede considerar que pasan las pruebas realizadas, cabe resaltar que el componente en general puede fallar en su objetivo de extraer y almacenar información como se muestra en la ilustración 21.

**Ilustración 21.** Resultado de las transacciones de extracción y almacenamiento.



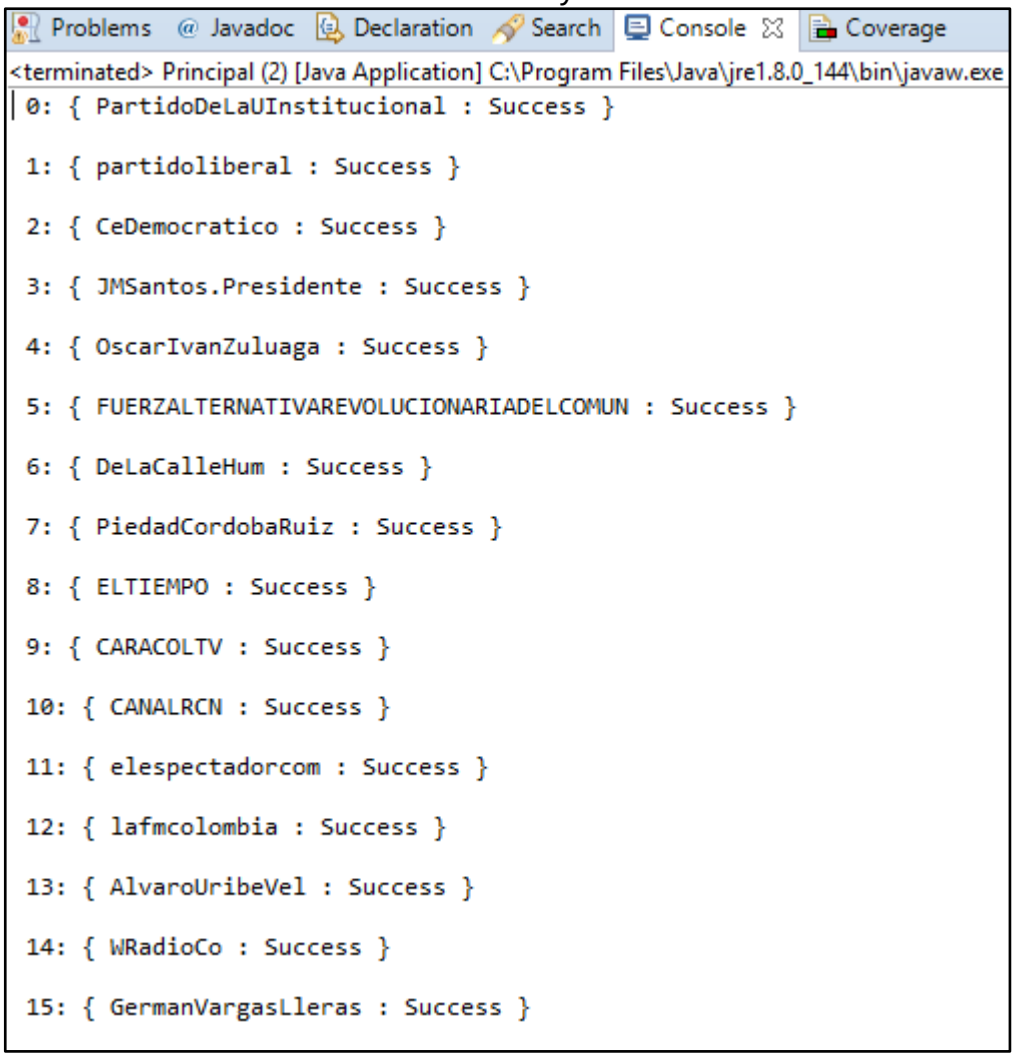

Fuente: Elaboración propia.

Las fallas en los procesos de extracción o almacenamiento se pueden dar por diferentes factores, sin embargo, el principal factor de fallas en el proceso de extracción esta dado porque el tiempo de respuesta de las peticiones de extracción se agota, cuando se solicita extraer un número muy elevado de publicaciones el tiempo de respuesta por parte del API de extracción aumenta, sin embargo, esto depende mucho de las condiciones de la infraestructura de red en las que se encuentre el componente, una petición con la misma cantidad de publicaciones a extraer puede fallar o no, dependiendo de la capacidad de respuesta de la red y el procesamiento del hardware, es por eso que en la sección *5.3* del *Anexo C* se hace una especificación de hardware óptimo para el adecuado funcionamiento del componente en cada nodo.

## 6.6 RESULTADOS

Dada por concluida la etapa de desarrollo e implementación del componente de extracción y almacenamiento, se presenta a continuación los resultados obtenidos después de realizadas las pruebas que evalúan la correcta funcionalidad del componente. Para la exposición de resultados se tomaron como referencia cuentas de la red social Facebook de medios de comunicación, personajes asociados a política en el país y grupos armados, sin embargo, la funcionalidad del componente desarrollado permite extraer información de un número de cuentas indefinido, en la ilustración 22 se muestra el resultado de extraer y almacenar información sobre quince cuentas seleccionadas en este contexto.

**Ilustración 22.** Transacciones de extracción y almacenamiento.

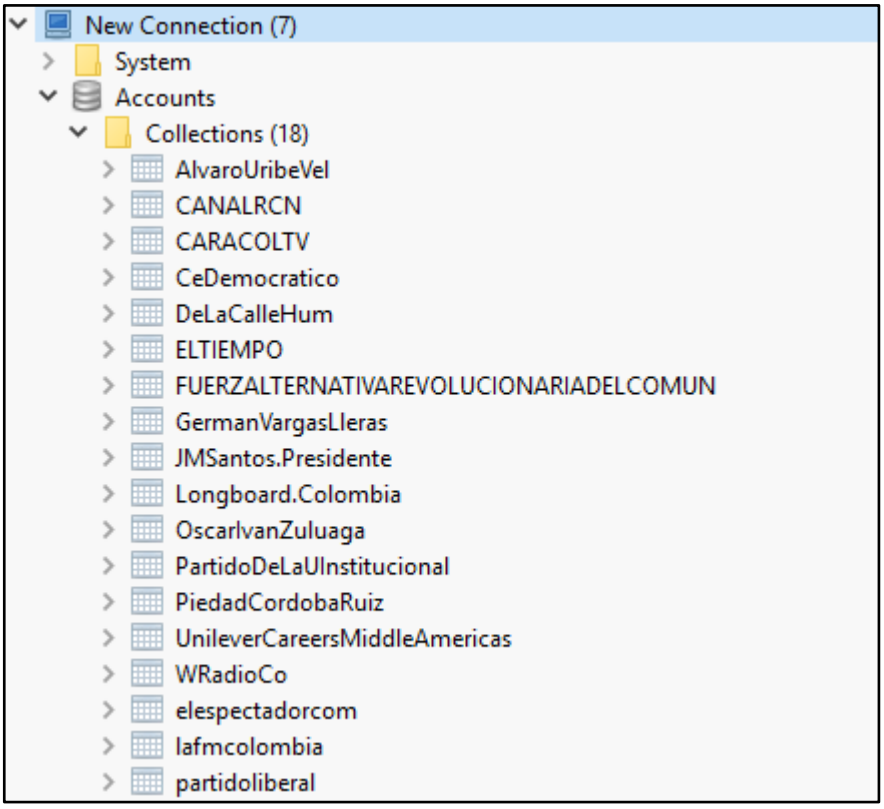



Fuente: Elaboración propia.

Una vez puesto en funcionamiento el componente, los procesos de extracción y almacenamiento se realizan por medio de la integración de funcionalidades y servicios entre todos los subcomponentes del proyecto, en la ilustración 22 se presentan quince cuentas seleccionadas sobre las cuales se extrajo y almacenó 2000 publicaciones por cada una ellas, al frente de ellas se observa el resultado de las transacciones.

La ilustración 22 permite analizar el resultado de las transacciones realizadas, sin embargo, para comprobar la información almacenada se debe usar una herramienta que permite ver de forma gráfica las colecciones creadas y los datos extraídos, esta herramienta es Robo3T. En la ilustración 23 se presentan las colecciones creadas en la base de datos no relacional MongoDb, una colección por cada cuenta sobre la que se hizo extracción.

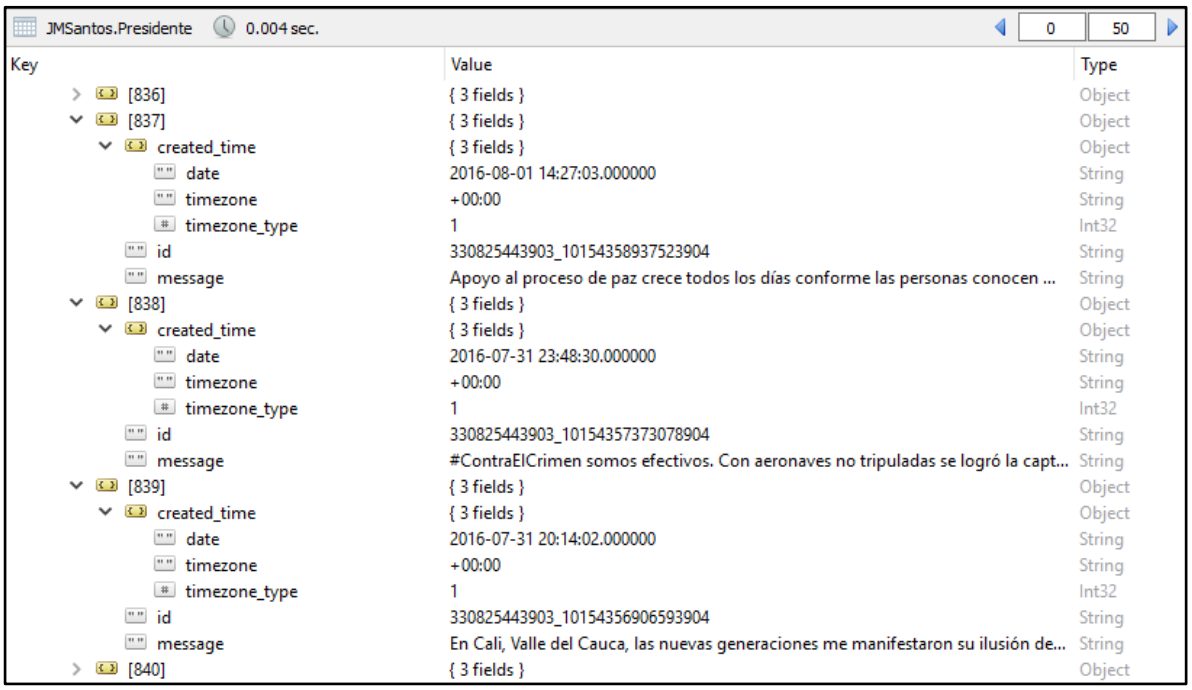
**Ilustración 23.** Colecciones de la base de datos “Accounts”.




Fuente: Elaboración propia.

Finalmente, en la ilustración 24 se observa la estructura de una de las publicaciones extraídas de la cuenta JMSantos.Presidente.

**Ilustración 24.** Publicaciones extraídas de la cuenta JMSantos.Presidente.

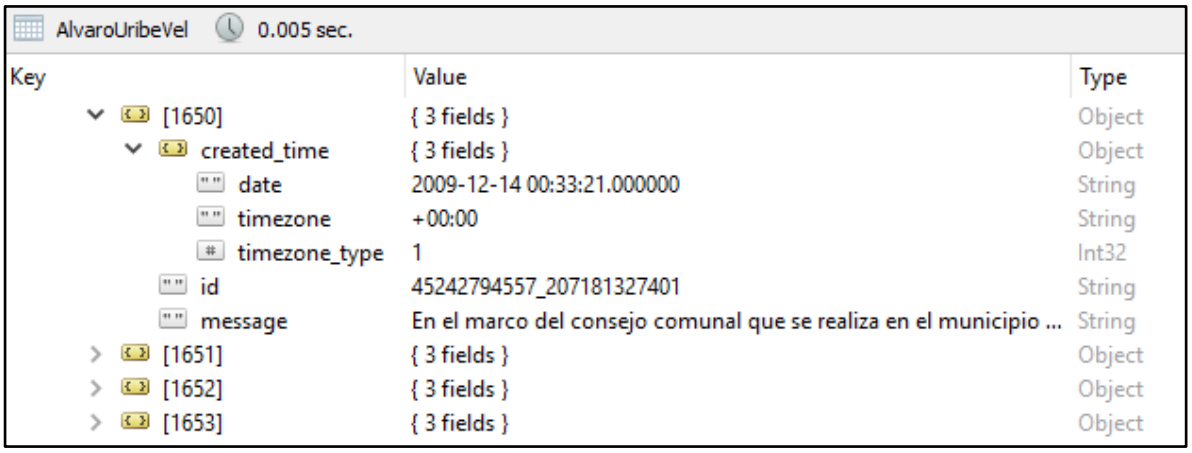


JMSantos.Presidente 0.004 sec. 0 50

| Key | Value | Type |
|---|---|---|
| [836] | { 3 fields } | Object |
| [837] | { 3 fields } | Object |
| created_time | { 3 fields } | Object |
| date | 2016-08-01 14:27:03.000000 | String |
| timezone | +00:00 | String |
| timezone_type | 1 | Int32 |
| id | 330825443903_10154358937523904 | String |
| message | Apoyo al proceso de paz crece todos los días conforme las personas conocen ... | String |
| [838] | { 3 fields } | Object |
| created_time | { 3 fields } | Object |
| date | 2016-07-31 23:48:30.000000 | String |
| timezone | +00:00 | String |
| timezone_type | 1 | Int32 |
| id | 330825443903_10154357373078904 | String |
| message | #ContraElCrimen somos efectivos. Con aeronaves no tripuladas se logró la capt... | String |
| [839] | { 3 fields } | Object |
| created_time | { 3 fields } | Object |
| date | 2016-07-31 20:14:02.000000 | String |
| timezone | +00:00 | String |
| timezone_type | 1 | Int32 |
| id | 330825443903_10154356906593904 | String |
| message | En Cali, Valle del Cauca, las nuevas generaciones me manifestaron su ilusión de... | String |
| [840] | { 3 fields } | Object |

Fuente: Elaboración propia.

**Ilustración 25.** Publicaciones extraídas de la cuenta AlvaroUribeVel.



AlvaroUribeVel 0.005 sec.

| Key | Value | Type |
|---|---|---|
| [1650] | { 3 fields } | Object |
| created_time | { 3 fields } | Object |
| date | 2009-12-14 00:33:21.000000 | String |
| timezone | +00:00 | String |
| timezone_type | 1 | Int32 |
| id | 45242794557_207181327401 | String |
| message | En el marco del consejo comunal que se realiza en el municipio ... | String |
| [1651] | { 3 fields } | Object |
| [1652] | { 3 fields } | Object |
| [1653] | { 3 fields } | Object |

Fuente: Elaboración propia.

El análisis de los resultados muestra que se logra extraer información de las cuentas seleccionadas anteriormente. Por otra parte, en comparación con herramientas existentes en el mercado actual se puede analizar que el presente proyecto resulta más potente dado que supera en capacidad y velocidad a algunas herramientas del mercado detalladas en los antecedentes.

# CONCLUSIONES

Una vez realizada la ejecución del proyecto y tomando como base los resultados obtenidos asociados a los procesos de extracción y almacenamiento de datos de la red social Facebook, se concluye que la información extraída corresponde y cumple con la calidad esperada para el componente, esto debido a que se extrae y se almacena la información sin alterar la integridad de los datos, esto permite formar una base sólida para el desarrollo de futuros trabajos, partiendo del uso de información verídica proporcionada por el desarrollo del presente proyecto.

El proceso de almacenamiento de información realizado en este proyecto plantea el reto de manejar diferentes tipos de datos, debido a la cantidad de estructuras y formatos de información que se pueden compartir según la naturaleza propia de la red social Facebook. De esta manera se concluye que la fase de investigación tuvo un papel muy importante porque permitió el acercamiento y uso de las tecnologías que más se adaptaron a los retos y necesidades descritas, para este caso lograr la integración de bases de datos no relacionales al proyecto que a su vez permitieron almacenar y manipular diferentes tipos de datos de forma ágil.

Teniendo en cuenta los resultados obtenidos en cada una de las fases durante el desarrollo del componente de extracción y almacenamiento de datos, se concluye que es importante respetar el proceso de análisis, diseño e implementación de una solución, no solo para optimizar el tiempo requerido en el desarrollo, sino evitar sobrecostos en el transcurso del proyecto, cabe resaltar que es de vital importancia realizar un adecuado proceso de levantamiento y definición de requerimientos, de esta manera se tienen bases sólidas para el diseño e implementación de un buen proyecto de desarrollo.

Según los resultados obtenidos, el uso de diferentes tecnologías permitió llevar a cabo la funcionalidad esperada para el componente de extracción y almacenamiento, mediante la implementación de servicios web que juegan un papel fundamental en la integración e intercambio de información entre las diferentes tecnologías utilizadas. Respecto a lo anterior se concluye que es posible y a veces necesario utilizar diferentes tecnologías en el desarrollo de un proyecto, la importancia recae en lograr hacer una adecuada integración entre estas, en busca de un mismo objetivo.

Debido al contexto tan amplio enmarcado en el presente proyecto, el desarrollo de éste permite el uso de diferentes tipos de herramientas y conocimientos necesarios en el campo de Ingeniería de Sistemas, permitiendo al desarrollador reforzar sus conocimientos y ampliar sus habilidades de análisis en el campo profesional. Por otra parte, teniendo en cuenta la descripción, el enfoque de la problemática a tratar y las limitaciones adjuntas al desarrollo del componente. Se concluye que los objetivos planteados para el desarrollo del presente proyecto fueron cumplidos,

abarcando los aspectos y las actividades necesarias para desarrollar el proyecto de acuerdo a su alcance, permitiendo de esta manera obtener la solución esperada por parte de los interesados al finalizar la etapa de implementación.

# RECOMENDACIONES

Se recomienda dar continuidad al desarrollo de diferentes proyectos que integren en general la funcionalidad del presente proyecto, dadas las múltiples alternativas de desarrollo que podrían surgir, como componentes de visualización que permitan apreciar de forma clara la información extraída o mejor aún la información extraída después de un proceso de análisis.

Como se ha indicado anteriormente, el desarrollo del presente proyecto está enmarcado en el ámbito de un proyecto de investigación de la Universidad Católica de Colombia, el componente de extracción y almacenamiento es uno de los varios componentes que se pueden desarrollar de forma adjunta al macroproyecto de investigación. El componente de extracción pretende ser una fuente de información de la cual se puede sacar mucho provecho para trabajos futuros.

En referencia a lo anterior, la importancia de tener accesibilidad a la información es que se pueden desarrollar proyectos que tomen como punto de partida la información extraída y almacenada del presente proyecto para poder realizar tareas de análisis, enfocado en un contexto especifico, con el análisis de la información se puede llegar a tener datos que sirvan para tomar decisiones estratégicas, no solo en el ámbito de violencia política, contexto en el cual se desarrolló el presente proyecto, sino además en el campo de inteligencia de negocios o cualquier otro contexto.

Una ventaja que se destacó en el desarrollo del proyecto fue la capacidad de integrar diferentes tecnologías por medio de servicios web, se recomienda de igual manera para desarrollos de trabajos futuros, tener en cuenta este tipo de tecnología de servicios web porque permite intercambiar información entre distintos lenguajes y aplicaciones, de esta manera se podría realizar una integración entre el componente de extracción y almacenamiento y algún otro desarrollo futuro.

# BIBLIOGRAFIA


Aggarwal, C. C. (2011). An introduction to social network data analytics. *Social network data analytics*, 1-15.

Aho, A. V, & Ullman, J. D. (1979). Universality of data retrieval languages. En *Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Principles of programming languages* (pp. 110-119).

An, M. D. A. (2008). Performance assessment. *Computer and Information Science*, *131*, 1.

Baumgartner, R., Gatterbauer, W., & Gottlob, G. (2009). Web data extraction system. En *Encyclopedia of Database Systems* (pp. 3465-3471). Springer.

Bray, T. (2014). The javascript object notation (json) data interchange format.

Breiger, R. L. (2004). *The analysis of social networks*. na.

Casallas, R. (2012). XP- EXTREME PROGRAMMING. Recuperado a partir de http://www.icesi.edu.co/departamentos/tecnologias_informacion_comunicaciones/cursos/09561/102/home/_media/unidad1/07-01-xp-1.pdf

Chawathe, S., Garcia-Molina, H., Hammer, J., Ireland, K., Papakonstantinou, Y., Ullman, J., & Widom., J. (1994). The TSIMMIS project: Integration of heterogenous information sources. *Proceedings of IPSJ Conference*, 7--18. Recuperado a partir de http://ilpubs.stanford.edu:8090/66/

D'\iaz, J., Schiavoni, A., Osorio, A., Amadeo, P., & Charnelli, E. (2012). Integraci{ó}n de plataformas virtuales de aprendizaje, redes sociales y sistemas acad{é}micos basados en Software Libre. Una experiencia en la Facultad de Inform{á}tica de la UNLP. *Universidad Nacional de la Plata*.

da Silva, A. S., & Teixeira, J. S. (s. f.). A Brief Survey of Web Data Extraction Tools.

del Busto, H. G., & Enr'\iquez, O. Y. (2012). Bases de datos NoSQL. *Revista Telem@ tica. Vol*, *11*(3), 21-33.

EUROPSIS, & GISIC. (2016). *Efectividad De Un Protocolo De Reexperimentación Emocional Y Mindfulness En Adultos Expuestos a Situaciones Traumáticas En Un Contexto De Violencia Política*. Universidad Católica de Colombia.

Facebook. (2016). Política de datos. Recuperado a partir de https://www.facebook.com/policy.php

Facebook. (2017). Post. Recuperado 29 de agosto de 2017, a partir de https://developers.facebook.com/docs/graph-api/reference/v2.10/post/

Ferrara, E., De Meo, P., Fiumara, G., & Baumgartner, R. (2014). Web data extraction, applications and techniques: A survey. *Knowledge-Based Systems*, *70*, 301-323.

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee, T. (1999). *Hypertext transfer protocol--HTTP/1.1*.

Foundation, T. A. S. (2017). CouchDB. Recuperado a partir de http://couchdb.apache.org/

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), 137-144.

Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., & Fdez-Riverola, F. (2014). Web scraping technologies in an API world. *Briefings in bioinformatics*, *15*(5), 788-797.

González, F. E., Bol'\ivar, I. J., & Vázquez, T. (2003). *Violencia pol{'\i}tica en Colombia: de la naci{ó}n fragmentada a la construcci{ó}n del Estado*. Centro de Investigaci{ó}n y Educaci{ó}n Popular.

Haas, S. W., & Grams, E. S. (2000). Readers, authors, and page structure: A discussion of four questions arising from a content analysis of Web pages. *Journal of the Association for Information Science and Technology*, *51*(2), 181-192.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. (M. Kamber, Ed.) (Third edit). Elsevier. Recuperado a partir de https://books.google.es/books?hl=es&lr=&id=pQws07tdpjoC&oi=fnd&pg=PP1&dq=Data+mining:+concepts+and+techniques&ots=tyMvY-oA0-&sig=_oIETuocVk6WSWVrbN958XQEn8o#v=onepage&q&f=false

Howard, P. E. N., Rainie, L., & Jones, S. (2001). Days and nights on the Internet: The impact of a diffusing technology. *American Behavioral Scientist*, *45*(3), 383-404.

Imielinski, T., Virmani, A., & Abdulghani, A. (1996). DataMine: Application Programming Interface and Query Language for Database Mining. En *KDD* (Vol. 96, p. 256).

Kosala, R., & Blockeel, H. (2000). Web mining research: A survey. *ACM Sigkdd Explorations Newsletter*, *2*(1), 1-15.

Kruchten, P. B. (1995). The 4+ 1 view model of architecture. *IEEE software*, *12*(6), 42-50.

Lappo, D. P. (2005). Extreme Programming for Solo Projects, 2-48.

Liddy, E. D., Paik, W., McKenna, M. E., Weiner, M. L., Edmund, S. Y., Diamond, T. G., … Snyder, D. L. (2000). User interface and other enhancements for natural language information retrieval system and method. Google Patents.

Lotfy, A. E., Saleh, A. I., El-Ghareeb, H. A., & Ali, H. A. (2016). A middle layer solution to support ACID properties for NoSQL databases. *Journal of King Saud University-Computer and Information Sciences*, *28*(1), 133-145.

Matsuo, Y., Mori, J., Hamasaki, M., Nishimura, T., Takeda, H., Hasida, K., & Ishizuka, M. (2007). POLYPHONET: an advanced social network extraction system from the web. *Web Semantics: Science, Services and Agents on the World Wide Web*, *5*(4), 262-278.

McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D. J., & Barton, D. (2012). Big data. *The management revolution. Harvard Bus Rev*, *90*(10), 61-67.

Melton, J., & Simon, A. R. (1993). *Understanding the new SQL: a complete guide*. Morgan Kaufmann.

Michael, K., & Miller, K. W. (2013). Big data: New opportunities and new challenges [guest editors'

introduction]. *Computer*, *46*(6), 22-24.

Mogotsi, I. C. (2010). Christopher d. manning, prabhakar raghavan, and hinrich sch{ü}tze: Introduction to information retrieval. Springer.

MongoDB, I. (s. f.). mongoDB. Recuperado 20 de marzo de 2017, a partir de https://www.mongodb.com/es

Nafr'\ia, I. (2007). *Web 2.0: El usuario, el nuevo rey de Internet*. Gesti{ó}n 2000.

Parker, S. (1987). Relational Database Technology. *Curator: The Museum Journal*, *30*(2), 124-130.

Prieto Espinoza, A., & Prieto Campos, B. (2005). *Conceptos de informatica.* Mc Graw Hill.

Rao, J., & Su, X. (2004). A survey of automated web service composition methods. En *SWSWPC* (Vol. 3387, pp. 43-54).

Redislabs. (2017). Redis. Recuperado 1 de mayo de 2017, a partir de https://redis.io/

Rieder, B. (2013). Studying Facebook via data extraction: the Netvizz application. En *Proceedings of the 5th annual ACM web science conference* (pp. 346-355).

Singhal, A. (2001). Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, *24*(4), 35-43.

Tabares, M. S., Pineda, J. D., & Barrera, A. F. (2008). Un patr{ó}n de interacci{ó}n entre diagramas de actividades UML y sistemas workflow. *Revista EIA*, (10).

Tang, J., Zhang, D., & Yao, L. (2007). Social network extraction of academic researchers. En *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on* (pp. 292-301).

Technology, N. I. of S. and. (2017a). NIST. Recuperado 1 de octubre de 2017, a partir de https://www.nist.gov/

Technology, N. I. of S. and. (2017b). TREC Overview. Recuperado 1 de octubre de 2017, a partir de http://trec.nist.gov/overview.html

Thelwall, M. (2001). A web crawler design for data mining. *Journal of Information Science*, *27*(5), 319-325.

Vaish, G. (2013). *Getting started with NoSQL*. Packt Publishing Ltd.

Wagh, K., & Thool, R. (2012). A comparative study of soap vs rest web services provisioning techniques for mobile host. *Journal of Information Engineering and Applications*, *2*(5), 12-16.

Wilson, R. E., Gosling, S. D., & Graham, L. T. (2012). A review of Facebook research in the social sciences. *Perspectives on psychological science*, *7*(3), 203-220.

Zuckerberg, Mark, Andreessen, M. (2017). Statistics. Recuperado 1 de mayo de 2017, a partir de https://newsroom.fb.com/company-info/

# GLOSARIO

**API:** Una API es el mecanismo más útil para conectar dos softwares entre sí para el intercambio de mensajes o datos en formato estándar como XML o JSON(Imielinski, Virmani, & Abdulghani, 1996).

**APACHE TOMCAT:** Es un servidor de aplicaciones libre, que implementa las tecnologías desarrolladas en la plataforma Java EE y permite ejecutar aplicaciones desarrolladas bajo este lenguaje.

**DATOS:** Un dato es una representación simbólica de un atributo o variable cuantitativa o cualitativa, los datos son números, letras o símbolos que describen objetos, condiciones o situaciones(Prieto Espinoza & Prieto Campos, 2005).

**DESPLIEGUE:** Proceso por el que una aplicación informática pasa a estar lista para utilizarse(Prieto Espinoza & Prieto Campos, 2005).

**HTTP:** El protocolo de transferencia de hipertexto es el protocolo usado en cada transacción que se ejecuta en la web. Java EE Es una plataforma de programación para desarrollar y ejecutar aplicaciones en lenguaje de programación java. Está enfocada para aplicaciones con grandes cantidades de transacciones y/o empresariales.

**INTEGRACIÓN:** Entrar a formar parte de una asociación o un grupo o a tomar parte en una actividad, incorporar una cosa en otra más amplia(D'\iaz, Schiavoni, Osorio, Amadeo, & Charnelli, 2012).

**JSON:** JavaScript Object Notation, es un formato mínimo y legible para estructurar datos. Se usa principalmente para transmitir datos entre un servidor y una aplicación final(Bray, 2014)

**TCP:** Es uno de los principales protocolos de internet. Se pueden usar conexiones TCP para crear conexiones entre dos nodos para enviar un flujo de datos.

**UML:** Es el lenguaje de modelado de sistemas más conocido y utilizado en el mundo.

**WEB SERVICE:** Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones(Rao & Su, 2004).


# COMPONENTE DE EXTRACCIÓN Y ALMACENAMIENTO DE DATOS DE UNA RED SOCIAL PARA UNA HERRAMIENTA WEB

**MILTON DANIEL REY SUÁREZ**

**ESPECIFICACIÓN DE REQUERIMIENTOS**

**DIEGO ALBERTO RINCÓN YÁÑEZ MCSc**

Contenido

# I. Introducción

## I.I Resumen

El análisis y especificación de requerimientos es una de las tareas más importantes en el ciclo de vida del desarrollo de software(Agut, s. f.), porque permite determinar los pasos a seguir al momento de desarrollar una nueva aplicación. Este documento presenta la especificación de requerimientos de software del sistema “Componente de extracción y almacenamiento de datos de una red social para una herramienta web” (CEA).

En este documento se presenta inicialmente la introducción de la especificación de requerimientos de software, permitiendo entender el propósito y el ámbito del sistema que se quiere desarrollar. Muestra una descripción general del sistema teniendo en cuenta varios aspectos como lo son las funciones que el producto debe realizar, las características de los usuarios, las restricciones, las suposiciones y dependencias entre otros aspectos importantes descritos en el documento.

En detalle se describen los requerimientos específicos del sistema, para ello se contemplan las interfaces, las funciones, los requisitos de rendimiento y de diseño, la usabilidad, seguridad y el soporte para cada uno de estos requerimientos.

Por último, se presentan las conclusiones a las que se llegan con el desarrollo del documento y recomendaciones para seguir el mismo.

## I.II Propósito

El objeto de esta especificación es definir de forma clara y precisa todas las funcionalidades y restricciones del componente de extracción y almacenamiento que se desea construir.

El documento va dirigido al equipo de desarrollo del componente, por tal motivo será el medio de comunicación entre las partes implicadas en el proyecto, esta especificación está sujeta a revisiones por parte del arquitecto de la solución, que se realizarán por medio de sucesivas versiones del documento, hasta alcanzar su aprobación por parte del mismo. Una vez aprobado servirá de base al equipo de desarrollo para la construcción del componente de extracción y almacenamiento.

### I.III Autores

| Nombre | Milton Daniel Rey Suárez |
|---|---|
| Rol | Desarrollador |
| Categoría Profesional | Estudiante de Pregrado |
| Contacto | *mdrey05@ucatolica.edu.co* |

### I.IV Ámbito del Sistema

El componente de extracción y almacenamiento de datos que tendrá como nombre "CEA", tiene como objetivo principal realizar la extracción y almacenamiento de datos de la red social Facebook.

Este componente se encargará exclusivamente de extraer datos y almacenarlos de tal manera que en un futuro puedan ser analizados, o tratados. Cabe resaltar que este componente no realiza ningún tipo de análisis o tratamiento sobre la información extraída, por tal motivo pretende conservar la integridad de los datos recolectados.

El principal beneficio que genera el desarrollo de este componente es la posibilidad de generar soluciones a necesidades en distintos campos del conocimiento y la industria por medio del análisis aplicado a los datos extraídos de una red social y que permiten obtener información clave que sirve de carácter predictivo a la hora de tomar de decisiones.

### I.V Definiciones, Acrónimos y Abreviaturas

#### I.V.I Definiciones

| News Feed | Formato de datos utilizado para proporcionar a los usuarios de un sitio web contenido actualizado con frecuencia(Hoadley, Xu, Lee, & Rosson, 2010). |
|---|---|

| | |
|---|---|
| URL | Cadena compacta que representa un recurso disponible a través de Internet. Estas cadenas se denominan "Localizadores uniformes de recursos"(Berners-Lee, Masinter, & McCahill, 1994). |
| Stakeholders | Personas, organizaciones o actores que participan en un proyecto(Donaldson & Preston, 1995). |
| Post | Una entrada individual en el feed de un perfil. El perfil puede ser un usuario, una página, una aplicación o un grupo(Facebook, 2017). |
| JSON | JavaScript Object Notation(Crockford, 2006) es un formato de intercambio de datos independiente del idioma. |

## I.V.II Acrónimos

| | |
|---|---|
| CEA | Componente de extracción y almacenamiento. |
| ERS | Especificación de requerimientos de Software. |
| RF | Requerimiento funcional. |
| RNF | Requerimiento no funcional. |

## I.V.III Abreviaturas

| | |
|---|---|
| API | Interfaz de programación de aplicaciones |

### I.VI Visión General del Documento

La estructura del documento consta de tres secciones principales, la primera de ellas realiza una introducción al mismo, proporcionando una visión general de éste, teniendo en cuenta el ámbito del sistema y algunas definiciones importantes para el entendimiento del documento.

En la segunda sección se realiza la descripción general del sistema, con el fin de conocer las funciones que éste debe realizar, las características de los usuarios, y los factores, restricciones, supuestos y dependencias que afectan su desarrollo.

Finalmente, en la tercera sección del documento se definen de manera detallada los requerimientos específicos que debe satisfacer el sistema, teniendo en cuenta aspectos de funcionalidad, usabilidad, disponibilidad, desempeño, seguridad, soporte, restricciones de diseño, componentes comprados, interfaces, entre otros. Esta especificación está estructurada siguiendo las normas definidas por el estándar IEEE 830.

### I.VII Alcance

El desarrollo del componente permitirá la extracción y almacenamiento limpio de datos de la red social Facebook, para llevar a cabo la implementación del componente de extracción y almacenamiento de datos se tiene un periodo de tiempo de 16 semanas, el proceso de desarrollo del componente de extracción de datos conllevará a un posterior desarrollo de un componente de almacenamiento en un sistema de base de datos.

## II. Descripción

### II.I Funciones del producto

Las funciones que debe realizar el componente se dividen en dos categorías principales que son extracción y almacenamiento.

#### II.I.I Extracción

- ✓ Se trata de realizar la extracción de los datos contenidos en publicaciones o posts, de páginas públicas de la red social Facebook. Estas publicaciones o posts tienen atributos específicos y pueden variar dependiendo el tipo de publicación. Los atributos que puede tener una publicación son los indicados en la Figura 1.

**Figura 1.** Tipos de atributos dentro de un post.

| Post | | | | |
|---|---|---|---|---|
| from | message_tags | promotable_id | targeting | privacy |
| icon | name | promotion_status | to | message |
| instagram_eligibility | object_id | properties | type | link |
| is_hidden | parent_id | shares | updated_time | place |
| is_instagram_eligible | permalink_url | source | | story_tags |
| is_published | picture | status_type | with_tags | story |

Fuente: Elaboración propia.

- ✓ Extraer la información de tal manera que pueda ser manipulada e intercambiada entre distintos lenguajes utilizados en el componente.

- ✓ Dada una fan page, y un número determinado de publicaciones, extraer la cantidad de publicaciones solicitadas sobre dicha página.
- ✓ Permitir la parametrización de las cuentas de las cuales se desea realizar la extracción de datos.

### II.I.II Almacenamiento

- ✓ Almacenar la información correspondiente a las extracciones hechas sin alterar la integridad de los datos.
- ✓ Guardar los datos de tal manera que no se presente redundancia cuando se hagan nuevas extracciones sobre las mismas cuentas.
- ✓ Realizar el proceso de extracción y almacenamiento cada semana, guardando únicamente las nuevas publicaciones.

## II.II Características de los Usuarios

El componente a construir va dirigido principalmente a usuarios con conocimientos en integración de sistemas y parametrización, esto debido principalmente a la característica de backend del mismo, además no tiene interfaces graficas definidas por lo que se requiere conocimiento en aspectos técnicos y de código para su adecuada ejecución.

## II.III Restricciones

- ✓ La implementación del componente se realizará con lenguajes de programación orientados a objetos Java y Php.
- ✓ El almacenamiento se realizará en el sistema de bases de datos NoSQL MongoDB.
- ✓ Los servidores que soporten el componente son Tomcat y Apache.
- ✓ Para el proceso de extracción se utilizará el API autorizado por Facebook “Graph API”.

### II.IV Suposiciones y Dependencias

El hardware requerido en los equipos portátiles o de escritorio para el funcionamiento adecuado del componente debe cumplir algunas características específicas que son:

- ✓ Disco duro con capacidad de almacenamiento de 1 Terabyte (TB).
- ✓ Memoria RAM mayor o igual a 4 Gigabyte (GB).
- ✓ Procesador con velocidad de procesamiento mayor o igual a 2 Giga Hertz (GHz).

Por otra parte, el equipo de desarrollo cuenta con las plataformas de desarrollo y las herramientas necesarias para el desarrollo del componente.

En cuanto al software requerido para los computadores, mínimo deben tener JDK para poder ejecutar la aplicación con éxito, y conexión a internet.

### II.V Requisitos futuros

Teniendo en cuenta la cantidad de información almacenada y la demanda de hardware para su adecuado procesamiento, podría requerirse una escalabilidad horizontal de la base de datos soportada en varios nodos.

### II. VI Metodología

El proceso de desarrollo del proyecto estará basado en la metodología de Programación Extrema (Casallas, 2012), esta es una metodología ágil de desarrollo de software. Las características de esta metodología permiten realizar una programación más organizada y simple teniendo en cuenta la definición de los objetivos específicos, y la identificación de los requisitos del sistema.

Una vez realizado el diseño del componente, se realiza la tercera etapa que consiste en la codificación de dicho componente. En este punto existen grandes ventajas utilizando la metodología de Programación Extrema ya que esta busca que el código sea sencillo y entendible, permitiendo un mejor ambiente de trabajo por parte del programador. A esta codificación se le aplicaran pruebas de aceptación, esta será la cuarta etapa y tendrá como objetivo encontrar posibles errores o

falencias para corregirlos, esto se puede conseguir haciendo re especificaciones y teniendo en cuenta las versiones que se van sacando del software.

La programación extrema es una metodología ágil para desarrollo de software basada en cuatro principios que son simplicidad, comunicación, feedback o retroalimentación y coraje.

Esta metodología está diseñada para ser usada en pequeños grupos de desarrollo, en el caso particular para una sola persona.

Ventajas de Programación Extrema

- Código valioso y más organizado en producción anticipada.
- Más fácil para el cliente cambiar su mente.
- Robusto conjunto de pruebas para todo el ciclo de vida.
- Vista precisa del estado del proyecto.
- Cierre de funciones y menor tasa de errores.

Desventajas de Programación Extrema

- Es recomendable emplearla solo en proyectos a corto plazo.
- Altas comisiones en caso de fallar.
- Puede no siempre ser más fácil que el desarrollo tradicional.

## III. Requerimientos Específicos

A continuación, se detallan los requisitos que el sistema deberá implementar.

| | |
|---|---|
| Código | RF-01 |
| Nombre de requerimiento | Autenticación y autorización. |
| Prioridad | Alta |

**Descripción**: El componente deberá permitir al usuario autenticarse para poder cumplir las funciones de extracción requeridas por el API de Facebook. Esto incluye brindar al sistema la autorización que permita obtener las publicaciones o posts de las cuentas deseadas.

**Entradas**: Usuario de Facebook y contraseña.

**Proceso**: Dado el usuario y la contraseña, el usuario se autentica por única vez, dando autorización al API de Facebook para que extraiga la información deseada por medio de la cuenta acreditada, generando un Token de acceso.

**Salidas**: Cadena de caracteres llamada “Token de acceso”.

| Código | RF-02 |
|---|---|
| Nombre de requerimiento | Extracción de publicaciones. |
| Prioridad | Alta |

**Descripción**: El componente permitirá extraer un número de publicaciones o posts de una página solicitada, siempre y cuando esta cuente con dicha cantidad de posts, de lo contrario extraerá el máximo de publicaciones dentro del rango solicitado, cabe resaltar que la página a extraer debe ser de carácter público. La extracción de datos de páginas o perfiles privados no se puede realizar.

**Entradas**: Página pública de Facebook, número de publicaciones a extraer.

**Proceso**: Dada la página a extraer y el número de publicaciones, el componente se encarga de llamar al API de Facebook indicándole estos dos parámetros y adicionalmente el token de acceso obtenido en la autenticación, si la transacción es exitosa se obtienen los datos de las publicaciones.

**Salidas**: Publicaciones requeridas en formato JSON.

| Código | RF-03 |
|---|---|
| Nombre de requerimiento | Almacenamiento de información. |
| Prioridad | Alta |

**Descripción**: El componente deberá ser capaz de almacenar la información obtenida en formato JSON, cuidando la no redundancia de los datos. si las publicaciones a almacenar ya se encuentran en la base de datos, estas se ignorarán, permitiendo almacenar solo aquellas publicaciones nuevas, o que no existan en la base de datos.

**Entradas**: Nombre de colección(página), JSON con la información extraída.

**Proceso**: Dado el nombre de la página y el JSON a almacenar, se envían estos datos a MongoDB, este crea una colección con el nombre de la página si no existe,

si existe, se guardan los datos extraídos, cabe resaltar que se guardaran únicamente aquellas publicaciones que no hayan sido guardadas antes.

**Salidas**: Estado de la transacción (Transacción exitosa, Transacción Fallida).

| | |
|---|---|
| Código | RF-04 |
| Nombre de requerimiento | Integración se servicios. |
| Prioridad | Alta |

**Descripción**: El componente deberá permitir la integración de todos los servicios, tanto de extracción como de almacenamiento y consultas. Permitiendo acceder a todas las funcionalidades del componente desde un solo lugar.

**Entradas**: Rutas de los servicios web.

**Proceso**: Dadas las rutas de acceso a los servicios web, el componente tendrá acceso a cada servicio, integrando todas las funciones, al ejecutar el componente, éste llamará a cada servicio y de esa manera lograr el cometido de extraer y almacenar la información de los posts.

**Salidas**: Componente integrado.

| | |
|---|---|
| Código | RF-05 |
| Nombre de requerimiento | Servicio SQL |
| Prioridad | Media |

**Descripción**: El componente permitirá consultar las cuentas de las que se quiere extraer información por medio de un servicio web, estas cuentas estarán almacenadas en una base de datos relacional, separadas por categorías (Medios de comunicación, personajes, partidos políticos).

**Entradas**: Nombre de base de datos, nombre de tabla.

**Proceso**: Dado el nombre de la base de datos y la tabla a consultar, el servicio web se encarga de consultar y retornar las cuentas sobre las cuales se quiere extraer información, lo anterior en formato JSON para una fácil manipulación entre lenguajes.

**Salidas**: Cuentas de las que se quiere extraer información en formato JSON, en caso de éxito, si no, mensaje de respuesta con código de petición http.

| Código | RF-06 |
|---|---|
| Nombre de requerimiento | Consulta información almacenada |
| Prioridad | Alta |

**Descripción**: EL componente permitirá consultar la información contenida en la base de datos NoSQL mediante un servicio web.

**Entradas**: Página o colección.

**Proceso**: Dada el nombre de la página o la colección, el servicio web consulta la base de datos y retorna la información contenida de la página solicitada.

**Salidas**: JSON de la información contenida en dicha colección de la base de datos.

Operación fallida.

| Código | RF-07 |
|---|---|
| Nombre de requerimiento | Servicio Php |
| Prioridad | Alta |

**Descripción**: El componente permitirá acceder al proceso de extracción de publicaciones por medio de un servicio web implementado en Php. Teniendo en cuenta la necesidad de integrar todas las funcionalidades del componente por medio de la tecnología de los servicios web.

**Entradas**: Ruta de acceso, página y número de publicaciones.

**Proceso**: El servicio web alojado en la ruta de acceso será capaz de llamar al proceso de extracción de publicaciones, pasándole como parámetros la página de la cual se quieren extraer datos y el número de publicaciones.

**Salidas**: Estado de la transacción (Transacción exitosa, Transacción Fallida).

| Código | RF-08 |
|---|---|
| Nombre de requerimiento | Servicio MongoDB |
| Prioridad | Media |

**Descripción**: El componente deberá acceder al proceso de almacenamiento de la información por medio de un servicio web, esto debido a la necesidad de integrar todas las funcionalidades del componente.

**Entradas**: Ruta de acceso, página, información en formato JSON.

**Proceso**: Dados los parámetros de entrada, el servicio web se encargará de utilizar el proceso de almacenamiento, al cual le enviará la información obtenida como parámetros y en retorno espera una respuesta de la transacción.

**Salidas**: Estado de la transacción (Transacción exitosa, Transacción Fallida).

| | |
|---|---|
| Código | RF-09 |
| Nombre de requerimiento | Actualización de información |
| Prioridad | Alta |

**Descripción**: El componente deberá ser capaz de actualizar la información contenida en la base de datos, esto quiere decir que se extraerán y almacenarán nuevos datos publicados en las cuentas o páginas indicadas periódicamente.

**Entradas**: Listado de páginas o cuentas a actualizar.

**Proceso**: Teniendo la lista de las cuentas sobre las cuales se quiere actualizar información, el componente llamara a los servicios correspondientes de extracción y almacenamiento para guardar la información de nuevas publicaciones periódicamente.

**Salidas**: Base de datos actualizada.

## IV. Disponibilidad

El sistema ha sido desarrollado teniendo en cuenta las reglas y requerimientos anteriormente mencionados, debido a la necesidad de actualizar cada semana el contenido y la información almacenada, el sistema deberá estar disponible al menos una vez a la semana, sin embargo, de ser requerida una actualización de información inesperada, el componente será capaz de hacerlo, teniendo los servicios disponibles el 95% del tiempo, el 5% del tiempo restante es para tareas administrativas sobre el sistema o de mantenimiento.

## V. Soporte

El componente cuenta con características parametrizables lo que permitirá futuros mantenimientos. El componente será integrado para que interactúe con otros componentes, durante este proceso es posible realizar un mantenimiento preventivo según las necesidades que se presenten.

## VI. Seguridad

La seguridad del componente está restringida principalmente por el uso de usuario y contraseña para autenticarse en él, cabe resaltar que la consistencia de esta información está a cargo del inicio de sesión de Facebook propiamente.

## VII. Interfaces

### VII.I Interfaz de comunicación Software

El componente de extracción y almacenamiento esta implementado en diferentes lenguajes de programación, adicionalmente cuenta con funcionalidades que son independientes, mostrando de esta manera la necesidad de comunicarse entre sí. La interfaz mediante la cual se comunicarán todas las funcionalidades del componente será por medio de servicios web, permitiendo la integración del componente. Estos se encontrarán en un servidor Apache y Tomcat y podrán ser llamados por los distintos procesos involucrados en el componente.

## VIII. Glosario

Colección – En MongoDb las colecciones almacenan datos relacionados, de igual forma contienen documentos que tienen otro nombre para los atributos.

Web Service - Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Servidor.- Computadora conectada a una red que pone sus recursos a disposición del resto de los integrantes de la red. Suele utilizarse para mantener datos centralizados o para gestionar recursos compartidos.

## IX. Bibliografia

Agut, R. M. (s. f.). Especificaci{ó}n de Requisitos Software seg{ú}n el est{á}ndar de IEEE 830. *Universidad Jaume I. Departamento de Inform{á}tica. Paper*.

Berners-Lee, T., Masinter, L., & McCahill, M. (1994). *Uniform resource locators (URL)*.

Casallas, R. (2012). XP- EXTREME PROGRAMMING. Recuperado a partir de http://www.icesi.edu.co/departamentos/tecnologias_informacion_comunicaciones/cursos/09561/102/home/_media/unidad1/07-01-xp-1.pdf

Crockford, D. (2006). The application/json media type for javascript object notation (json).

Donaldson, T., & Preston, L. E. (1995). The stakeholder theory of the corporation: Concepts, evidence, and implications. *Academy of management Review*, *20*(1), 65-91.

Facebook. (2017). Post. Recuperado 29 de agosto de 2017, a partir de https://developers.facebook.com/docs/graph-api/reference/v2.10/post/

Hoadley, C. M., Xu, H., Lee, J. J., & Rosson, M. B. (2010). Privacy as information access and illusory control: The case of the Facebook News Feed privacy outcry. *Electronic commerce research and applications*, *9*(1), 50-60.


# COMPONENTE DE EXTRACCIÓN Y ALMACENAMIENTO DE DATOS DE UNA RED SOCIAL PARA UNA HERRAMIENTA WEB

**MILTON DANIEL REY SUÁREZ**

**DOCUMENTO DISEÑO DE SOFTWARE**

**DIEGO ALBERTO RINCÓN YÁÑEZ MCSc**

Contenido

# Tabla de Ilustraciones



# Tabla de Tablas

# I. Introducción

## I.I Alcance

El presente documento describe el diseño detallado del componente de extracción y almacenamiento de datos. El alcance de este documento pretende definir de forma clara la base para la codificación del componente, abarcando aspectos de diseño en alto nivel y bajo nivel del sistema, teniendo en cuenta restricciones, supuestos y dependencias, así como la arquitectura general. Empleando diferentes vistas de componentes, despliegue y secuencia.

## I.II Supuestos y dependencias

En esta sección se describen los supuestos identificados para el diseño adecuado del componente, agrupados en tres grupos generales que son, equipo de desarrollo, equipos y cliente.

| Grupo | Suposiciones y dependencias |
|---|---|
| Equipo de Desarrollo | El equipo de desarrollo cuenta con las herramientas y plataformas necesarias para el desarrollo adecuado del componente de extracción y almacenamiento, por ejemplo, entornos de desarrollo integrado. |
| Equipos (Computadores) | Los equipos en los cuales se ejecutará el componente cumplen con el hardware requerido el cual está descrito en el inciso *II.IV Suposiciones y Dependencias*, del documento de especificación de requerimientos de software SRS.<br><br>En cuanto a el software requerido, como mínimo debe tener instalado JDK para la ejecución del componente. |

| | |
|---|---|
| Cliente | No se realizarán modificaciones sobre los requerimientos funcionales del sistema durante el desarrollo del componente, dado el tiempo limitado para el desarrollo. |

*Tabla1. Suposiciones y dependencias.*

## I.IV Restricciones

### I.IV.I Restricciones Generales

- ✓ El componente estará desarrollado únicamente en el idioma español.
- ✓ Tiempo de entrega del proyecto es de máximo 16 semanas.

### I.IV.II Restricciones de Software

- ✓ La implementación del componente se realizará con lenguajes de programación orientados a objetos Java y Php.
- ✓ El almacenamiento se realizará en los sistemas de bases de datos NoSQL MongoDB y SQL MySQL.
- ✓ Los servidores que soporten el componente son Tomcat y Apache.
- ✓ Para el proceso de extracción se utilizará el API autorizado por Facebook “Graph API”.

### I.IV.III Restricciones de Hardware

En la sección II.IV del documento de especificación de requerimientos de software(SRS) están descritas las características de hardware necesarias para el desarrollo y ejecución del componente.

## I.V Riesgos

A continuación, se describen los riesgos que están relacionados con el componente de extracción y almacenamiento:

- ✓ El tamaño general del proyecto se ha subestimado.
- ✓ El entorno de desarrollo, asociado con la disponibilidad y calidad de las herramientas que se van a emplear en la construcción del componente no es el adecuado.
- ✓ El diseño del componente no es el adecuado, requiere más tiempo afectando el cronograma de desarrollo.
- ✓ Pobre o nulo levantamiento de requerimientos para el desarrollo adecuado del componente.
- ✓ El tiempo requerido para desarrollar el software no es el estimado debido a su tamaño.
- ✓ Metodología inadecuada para el desarrollo del proyecto, se elige la que se cree conveniente, sin embargo, no resulta ser la esperada.

## I.VI Metodología de Pruebas

Se realizan diferentes tipos de pruebas a lo largo del desarrollo del proyecto con el objetivo de encontrar defectos, ratificar la confianza y calidad del componente y evitar la aparición de defectos o errores inesperados, a continuación, se describe las pruebas realizadas al componente de extracción.

Evaluación estática, pretende detectar manualmente defectos en cualquier producto del desarrollo, quiere decir, que el producto en cuestión (sea requisito, diseño, código, etc.) es analizado mediante la lectura del mismo, sin ejecutarlo (Febles Estrada et al., 2011).

Prueba del sistema y prueba de aceptación. Ambas involucran pruebas sobre el programa total. La primera intenta verificar que el programa cumple con las

especificaciones y se hace con datos de prueba. La segunda se hace con datos reales y prueba el programa contra los requerimientos. Es prácticamente una validación(de Abadia, 2010).

Prueba de Integración que consiste en probar la estructura del programa y el algoritmo que el programa implementa.

Prueba funcional que efectúa la verificación de una función dada del programa.

Prueba de módulo o unidad, pone a prueba la lógica de los módulos como unidad solamente, sin importar su función dentro del programa global.

# II. Arquitectura del Sistema

## II.I Nivel General

De acuerdo con los aspectos especificados en el documento de requerimientos SRS, la arquitectura del componente general estará diseñada de tal manera que atienda las funcionalidades en cuanto a las características de extracción y almacenamiento. El componente concentrará sus módulos en un solo computador de manera centralizada, sin embargo, cabe aclarar que esto es independiente de la interacción con las bases de datos del componente que podrían estar alojadas remotamente gracias a las interfaces de comunicación mediante servicios web.

La arquitectura del sistema estará organizada en tres capas principales, en cada una de ellas se alojarán los componentes de acuerdo a la función que cumplen, dichas capas son: Capa de integración, servicios y datos.

De acuerdo a lo anterior, la primera capa es la de integración, en un nivel superior es la encargada de interactuar con la capa de servicios, se encarga de organizar el proceso del llamado a los servicios que comprenden la funcionalidad general del componente obteniendo los parámetros de entrada y salida de cada servicio.

En segundo lugar, se encuentra la capa de servicios, es la capa responsable del core del componente porque en ella se encuentran las funcionalidades principales del mismo, compuesta por servicios web que se encargan por separado de realizar tareas fundamentales como autenticación, extracción y almacenamiento respectivamente.

Finalmente, se encuentra la capa de datos, es la encargada de almacenar la información total del componente, esta interactúa con la capa de servicios.

## II.I.I Diagrama de Componentes

En la siguiente ilustración, se muestra en detalle cómo están separados los componentes que conforman el componente de extracción y almacenamiento, y en que niveles se encuentra cada uno, de acuerdo a las funcionalidades que tenga destinadas realizar. El concepto de niveles o capas es aplicable para dar una idea clara de cómo está organizado el componente en general, aunque no quiere decir que cada una se encuentre operando en equipos o servidores diferentes a través de la web.

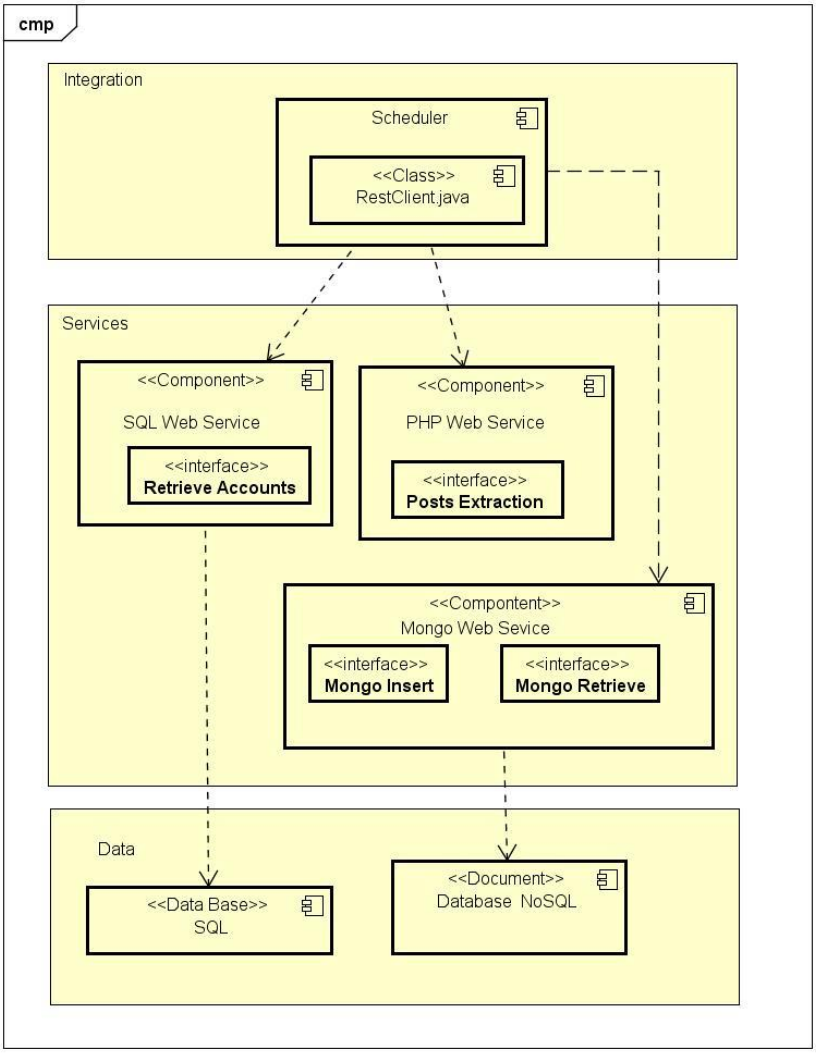



*Ilustración 1. Diagrama de componentes.*

*Scheduler:* Este componente es el encargado de integrar toda la funcionalidad del "Componente de extracción y almacenamiento", interactuando con los servicios web, permitiendo la ejecución de manera ordenada de los diferentes componentes que hacen parte del sistema.

*SQL Web Service:* Es el componente encargado de consultar las llaves de autenticación y las cuentas sobre las cuales se quiere extraer información, esta información será utilizada respectivamente como las entradas del componente de extracción php.

*PHP Web Service:* Además de comunicarse con el Scheduler, este componente puede considerarse como el corazón del sistema debido a su importancia, es el encargado de la extracción de las publicaciones o posts de la red social Facebook. Por medio del API(Imielinski, Virmani, & Abdulghani, 1996) "Graph Api" y de las credenciales de acceso obtenidas por el componente SQL extrae los datos de las cuentas requeridas en formato Json y los retorna para que puedan ser almacenados respectivamente.

*Mongo Web Service:* Se encarga de gestionar el almacenamiento de la información entrante por parte del componente de extracción Php y consultar la información contenida en la base de datos NoSQL, por medio de dos interfaces llamadas Mongo Insert y Mongo Retrieve, ambas interfaces intercambian información en formato Json.

*SQL Database:* Este componente se encuentra en la capa de datos de la arquitectura del componente general, se encarga de almacenar las llaves de acceso al API de extracción, de igual manera, contiene las cuentas de las páginas de las que se extraerán los posts o publicaciones.

*NoSQL Database:* Este componente al igual que el anterior, se encuentra en la capa de datos, es el encargado de almacenar toda la información extraída por parte del componente Php, es NoSQL por la ventaja en el almacenamiento de grandes cantidades de información en comparación a una base de datos relacional.

## II.I.II Diagrama de Despliegue

Los diagramas de despliegue muestran la configuración de los nodos que participan en la ejecución del componente y de los componentes que los conforman. En la ilustración 2 se muestra el diagrama de despliegue del componente de extracción y almacenamiento. Para este caso observamos que es posible realizar el despliegue completo en una sola máquina que soportaría todo el sistema, siguiendo las especificaciones de este nodo (ver *Tabla 2*).

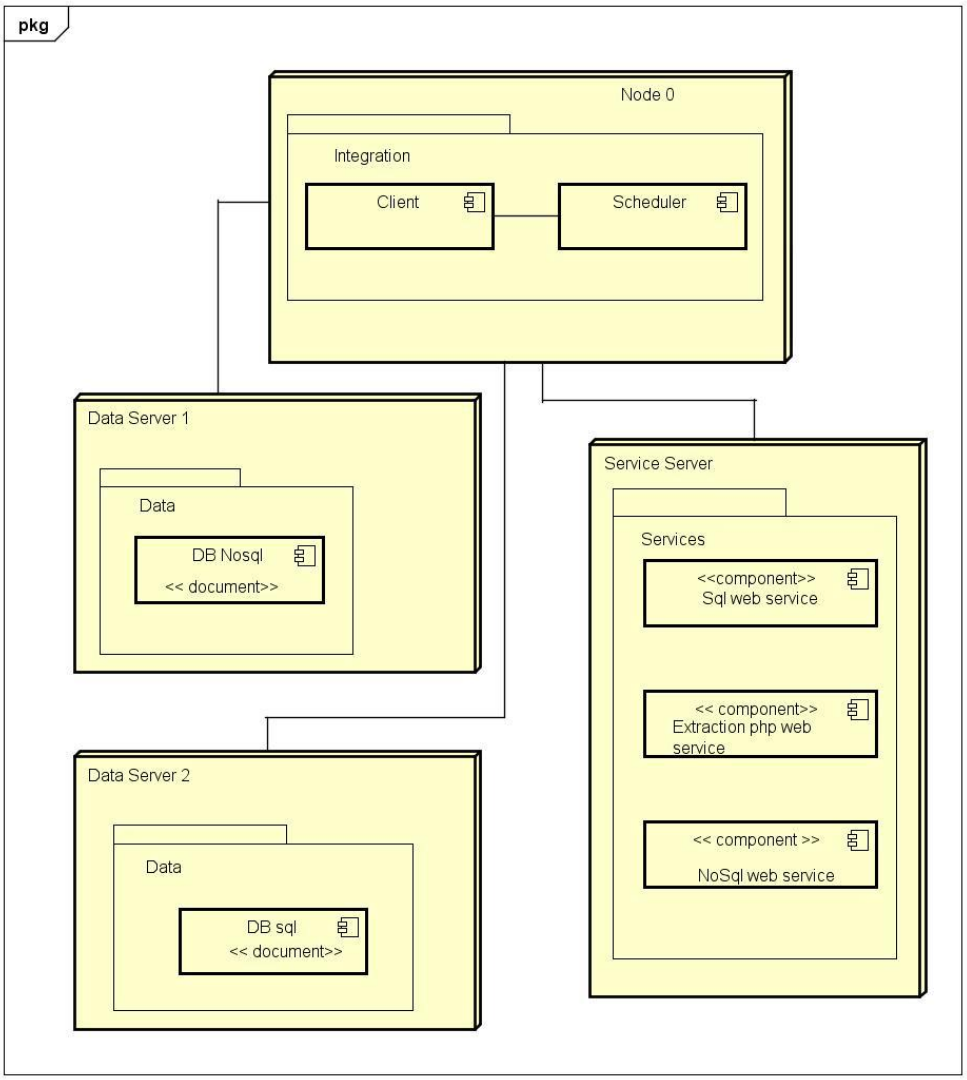



*Ilustración 2. Diagrama de despliegue. Fuente: Elaboración propia.*

| Nombre | Nodo de despliegue 0 |
|---|---|
| Especificación | ✓ Disco duro de 1 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 2Gz o superior. |
| Capas | Integración |

*Tabla 2. Especificación de Nodo de Despliegue*

| Nombre | Data Server 1 |
|---|---|
| Especificación | ✓ Disco duro de 10 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Apache HTTP Server<br>✓ Memoria RAM 16 Gb o superior |
| Capas | Datos |

*Tabla 3. Especificación de Data Server 1*

| Nombre | Data Server 2 |
|---|---|
| Especificación | ✓ Disco duro de 10 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Memoria RAM 16 Gb o superior<br>✓ Apache Tomcat V7 |
| Capas | Datos |

*Tabla 4. Especificación de Data Server 2*

| Nombre | Servicie Server |
|---|---|
| Especificación | ✓ Disco duro de 1 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Memoria RAM 16 Gb o superior |
| Capas | Servicios |

*Tabla 5. Especificación de Service Server*

## II.I.III Diagrama de actividades

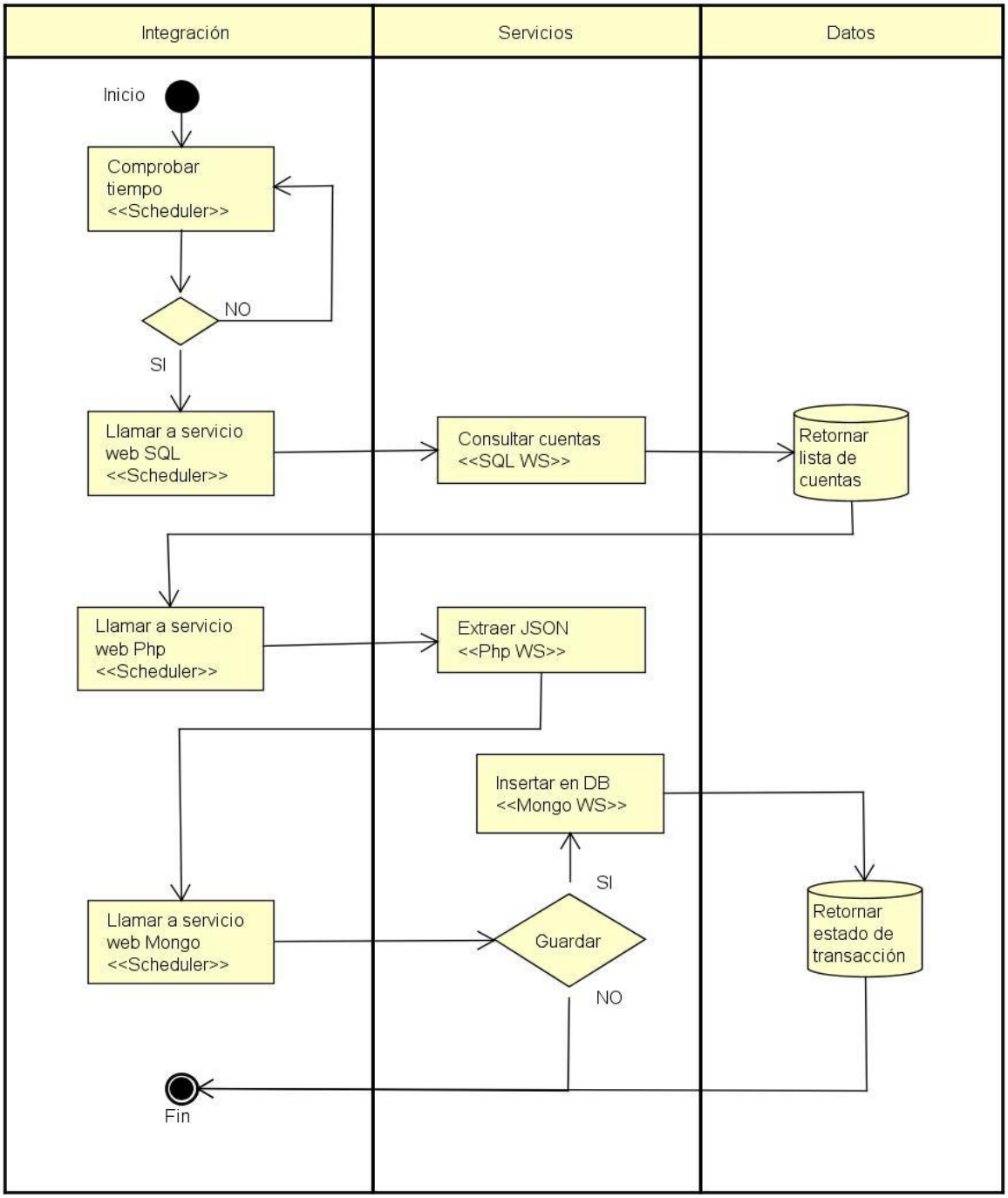



*Ilustración 3. Diagrama de actividades. Fuente: Elaboración propia.*

Los diagramas de comportamiento permiten tener un mejor entendimiento de lo que sucede cuando el componente realice alguna funcionalidad específica. Los diagramas de actividad tienen como propósito ilustrar el modo en el que los objetos de un sistema interactúan y el flujo de actividades realizadas por diferentes actores(Tabares, Pineda, & Barrera, 2008).

En la ilustración 3 se muestra el diagrama de actividades realizados para el desarrollo del componente de extracción y almacenamiento.

### II.II Sub-Arquitecturas de Componentes

Los componentes especificados en la arquitectura general presentada en la sección II.I del presente documento tienen aspectos importantes a considerar para su adecuada implementación.

El componente de extracción de datos llamado Php Web Service tiene una interfaz de comunicación mediante la cual intercambia información en formato Json, este servicio es consumido por un cliente Java que es llamado por el componente organizador o scheduler.

Por otra parte, el componente de almacenamiento de información llamado Mongo web service, tiene dos interfaces mediante las cuales intercambia información, la primera se llama Mongo Insert, es la encargada de recibir los datos de las publicaciones extraídas por parte del componente Php y llamar al componente de base de datos no relacional para hacer el respectivo almacenamiento. La segunda interfaz de comunicación se llama Mongo Retrieve, es la encargada de consultar la información de la base de datos no relacional, y devuelve tanto la información como el resultado de la transacción.

El componente de consulta de llaves llamado Sql web service tiene una interfaz mediante la cual accede a la información que se encuentra en una única tabla sql, dicha tabla contiene la información de las cuentas sobre las cuales se realiza la extracción de datos.

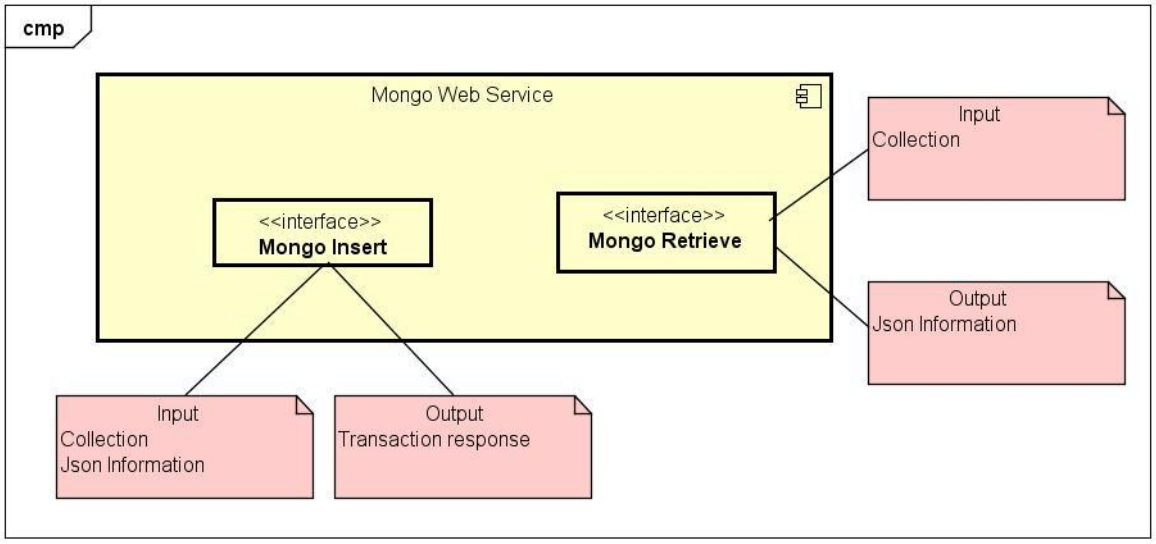



*Ilustración 4. Mongo web Service. Fuente: Elaboración propia.*

## II.II.I Responsabilidades

| Rol | Tareas | Responsable |
|---|---|---|
| Líder de proyecto | Gestionar las prioridades, mantener al equipo del proyecto enfocado en los objetivos. Supervisar el establecimiento de la arquitectura del sistema, planificación y control del proyecto. | Diego Rincón |
| Diseñador | Especificar y validar los requerimientos del sistema, elaboración del modelo de diseño, diseño de pruebas funcionales sobre el sistema. | Daniel Rey |
| Analista de requerimientos | Elaboración de la documentación del componente de extracción y almacenamiento, elaboración de los modelos de implementación y de despliegue. | Daniel Rey |

| | | |
|---|---|---|
| Desarrollador | Implementar la solución teniendo en cuenta los requisitos y requerimientos del sistema. Escribir, depurar y mantener el código del componente. | Daniel Rey |

*Tabla 6. Roles y responsabilidades.*

## II.II.III Modelado Base de Datos

Teniendo en cuenta la solución escogida de bases de datos NoSQL para el desarrollo del componente por su desempeño en las transacciones de grandes cantidades de información respecto de las bases de datos relacionales, se presenta un modelo de datos en el que la información es almacenada en formato Json explotando esta gran ventaja que brinda MongoDB (ver Ilustración 5) .


```
{
  "created_time": "2017-10-24T12:38:03+0000",
  "message": "La buena vida: Aprenda a diferenciar los productos realmente saludables para el organismo.",
  "story": "El Tiempo is feeling interesado.",
  "id": "148349507804_10154914990687805"
},
{
  "created_time": "2017-10-24T12:30:01+0000",
  "message": "Con tributos a Soda, Miguel Mateos y Los Prisioneros, recuerdan este hito del rock bogotano."
  "id": "148349507804_10154914434562805"
},
```


*Ilustración 5. Formato Json de publicación o post.*

# III. Políticas y Tácticas

Para lograr un desarrollo adecuado, correcto y completo del componente, se deben seguir una serie de políticas y tácticas que buscan guiar y establecer los lineamientos mediante los cuales se ejecutara el desarrollo del sistema.

## III.I Generales

Se debe estandarizar el ciclo de desarrollo del sistema, tal como lo establece la metodología de desarrollo.

Se debe planificar las etapas de desarrollo incluyendo recursos, criterios de aceptación, respaldos.

Los programadores tendrán acceso a la información necesaria inclusive si ésta contiene datos sensibles para la producción del componente.

Toda modificación de software debe ser analizada previamente en los ambientes de desarrollo y prueba.

### III.II Vulnerabilidades

Se debe efectuar validaciones y evaluaciones periódicas sobre la seguridad del componente durante el ciclo de vida del proyecto.

### III.III Documentación

El programador incluirá comentarios en el programa fuente, éstos deben ser útiles para un tercero sin divulgar información innecesaria.

La documentación se debe generar durante el ciclo de desarrollo y realizarla hasta el final.

Actualizar la documentación en caso que alguna funcionalidad del componente cambie.

## IV. Glosario

API - Una API es el mecanismo más útil para conectar dos softwares entre sí para el intercambio de mensajes o datos en formato estándar como XML o JSON(Imielinski et al., 1996).

Web Service - Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones(Rao & Su, 2004).

Integración - Entrar a formar parte de una asociación o un grupo o a tomar parte en una actividad, incorporar una cosa en otra más amplia(D'\iaz, Schiavoni, Osorio, Amadeo, & Charnelli, 2012).

Datos - Un dato es una representación simbólica de un atributo o variable cuantitativa o cualitativa, los datos son números, letras o símbolos que describen objetos, condiciones o situaciones(Prieto Espinoza & Prieto Campos, 2005).

Despliegue - Proceso por el que una aplicación informática pasa a estar lista para utilizarse(Prieto Espinoza & Prieto Campos, 2005).

Json - JavaScript Object Notation, es un formato mínimo y legible para estructurar datos. Se usa principalmente para transmitir datos entre un servidor y una aplicación final(Bray, 2014).

## V. Bibliografía y referencias

Bray, T. (2014). The javascript object notation (json) data interchange format.

D'\iaz, J., Schiavoni, A., Osorio, A., Amadeo, P., & Charnelli, E. (2012). Integraci{ó}n de plataformas virtuales de aprendizaje, redes sociales y sistemas acad{é}micos basados en Software Libre. Una experiencia en la Facultad de Inform{á}tica de la UNLP. *Universidad Nacional de la Plata*.

de Abadia, M. E. V. (2010). Procedimientos para prueba de software. *Publicaciones Icesi*, (18).

Febles Estrada, A., Capote Garc'\ia, T., León Perdomo, Y., Velázquez Cintra, A., Delgado Mart'\inez, R., & Calzadilla D'\iaz, R. (2011). Una experiencia novedosa para el testing desarrollada por un departamento de pruebas de software. *Revista Cubana de Ciencias Inform{á}ticas*, *5*(2).

Imielinski, T., Virmani, A., & Abdulghani, A. (1996). DataMine: Application Programming Interface and Query Language for Database Mining. En *KDD* (Vol. 96, p. 256).

Prieto Espinoza, A., & Prieto Campos, B. (2005). *Conceptos de informatica.* Mc Graw Hill.

Rao, J., & Su, X. (2004). A survey of automated web service composition methods. En *SWSWPC* (Vol. 3387, pp. 43-54).

Tabares, M. S., Pineda, J. D., & Barrera, A. F. (2008). Un patr{ó}n de interacci{ó}n entre diagramas de actividades UML y sistemas workflow. *Revista EIA*, (10).


# COMPONENTE DE EXTRACCIÓN Y ALMACENAMIENTO DE DATOS DE UNA RED SOCIAL PARA UNA HERRAMIENTA WEB

**MILTON DANIEL REY SUÁREZ**

**DOCUMENTO DE ARQUITECTURA DE SOFTWARE**

**DIEGO ALBERTO RINCÓN YÁÑEZ MCSc**

## CONTENIDO

## Tabla de Ilustraciones



## Tabla de Tablas

# 1. INTRODUCCIÓN

Esta sección expone una visión general del documento de arquitectura de software, abarcando el propósito, alcance, definiciones, acrónimos, abreviaturas y organización del documento.

## 1.1 Propósito

El propósito del presente documento es definir la arquitectura de software a implementar en el desarrollo del componente de extracción y almacenamiento de datos de una red social para una herramienta web, en él se presenta en términos de arquitectura y diseño, los componentes y requerimientos definidos en el Documento de Especificación de Requerimientos de software.

El presente documento es el punto de partida para empezar a realizar el proceso de desarrollo e implementación del componente de extracción.

Con referencia a lo anterior, el documento de arquitectura de software va dirigido principalmente a los stakeholders o a las personas involucradas en el proceso de desarrollo del componente, entre ellos se encuentra el analista de requerimientos, arquitecto de la solución, equipo de desarrollo y pruebas.

## 1.2 Alcance

El presente documento describe la arquitectura de software del componente de extracción y almacenamiento de datos mediante vistas del sistema. El alcance de este documento es definir de forma clara y precisa la base para la codificación del componente, abarcando aspectos de diseño del sistema. Empleando diferentes vistas del sistema como vistas estructurales, lógicas, de despliegue entre otras.

Adicionalmente, se direccionan otros aspectos que son importantes en el desarrollo de componente referentes a los requerimientos no funcionales que se buscan en el adecuado funcionamiento del componente.

Por otra parte, se definen los módulos en los que se divide la arquitectura del sistema y la relación que existe entre cada módulo para el funcionamiento global del componente. El documento de Arquitectura de Software hace parte de los entregables del proyecto de Componente de extracción y almacenamiento de datos de una red social para una herramienta web.

## 1.3 Definiciones, Acrónimos y Abreviaciones

Las palabras desconocidas o el vocabulario demasiado técnico utilizado en el presente documento, será definido a continuación en la tabla de definiciones acrónimos y abreviaciones.

| | |
|---|---|
| News Feed | Formato de datos utilizado para proporcionar a los usuarios de un sitio web contenido actualizado con frecuencia(Hoadley, Xu, Lee, & Rosson, 2010). |
| URL | Cadena compacta que representa un recurso disponible a través de Internet(Berners-Lee, Masinter, & McCahill, 1994). Estas cadenas se denominan "Localizadores uniformes de recursos". |
| Stakeholders | Personas, organizaciones o actores que participan en un proyecto. |
| Post | Una entrada individual en el feed de un perfil. El perfil puede ser un usuario, una página, una aplicación o un grupo(Facebook, 2017). |
| JSON | JavaScript Object Notation es un formato de intercambio de datos independiente del idioma(Bray, 2014). |
| CEA | Componente de extracción y almacenamiento. |
| ERS | Especificación de requerimientos de Software. |
| RF | Requerimiento funcional. |
| RNF | Requerimiento no funcional. |
| | Interfaz de programación de aplicaciones |

| | |
|---|---|
| API | |
| HTTP | El protocolo de transferencia de hipertexto es el protocolo usado en cada transacción que se ejecuta en la web(Fielding et al., 1999). |

Tabla 1. Definiciones, acrónimos y abreviaciones

## Referencias

Berners-Lee, T., Masinter, L., & McCahill, M. (1994). *Uniform resource locators (URL)*.

Bray, T. (2014). The javascript object notation (json) data interchange format.

Facebook. (2017). Post. Recuperado 29 de agosto de 2017, a partir de https://developers.facebook.com/docs/graph-api/reference/v2.10/post/

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee, T. (1999). *Hypertext transfer protocol--HTTP/1.1*.

Hoadley, C. M., Xu, H., Lee, J. J., & Rosson, M. B. (2010). Privacy as information access and illusory control: The case of the Facebook News Feed privacy outcry. *Electronic commerce research and applications*, *9*(1), 50-60.

Imielinski, T., Virmani, A., & Abdulghani, A. (1996). DataMine: Application Programming Interface and Query Language for Database Mining. En *KDD* (Vol. 96, p. 256).

Kruchten, P. B. (1995). The 4+ 1 view model of architecture. *IEEE software*, *12*(6), 42-50.

Tu, Q., & Godfrey, M. W. (2001). The build-time software architecture view. En *Software Maintenance, 2001. Proceedings. IEEE International Conference on* (pp. 398-407).

### 1.4 Organización del Documento

La organización del documento está dada para entender el marco en el que se desarrolla el proyecto. El documento de arquitectura de software está distribuido en secciones de la siguiente manera. En primer lugar, el documento presenta la introducción del mismo, en esta se realiza una descripción sobre el alcance y propósito, haciendo énfasis en las razones que hacen necesario el desarrollo del presente documento y hacia quien va dirigido, en esta sección también se define el vocabulario técnico o palabras ambiguas que se encuentran a lo largo del documento y pretende guiar al lector a un mejor entendimiento del mismo. En segundo lugar, se encuentra la representación de los stakeholders que hace una descripción de los actores involucrados en el desarrollo del proyecto, posteriormente se hace la descripción de la arquitectura que maneja el componente de extracción mediante el modelo 4 +1, éste modelo permite abordar la arquitectura basado en el uso de diferentes vistas. En tercer lugar, se hace la definición de los puntos de vista desde los cuales se analiza el desarrollo del proyecto, en este punto se define los puntos de vista de proceso, vista física, vista lógica y la vista de desarrollo. Seguidamente se presenta cada vista por separado, realizando la descripción de la misma en el proyecto mediante el uso de los diagramas que aplican para cada vista, siguiendo el lenguaje de modelado unificado (UML). En último lugar, se presenta el glosario y las referencias del documento.

## 2. REPRESENTACIÓN DE STAKEHOLDERS

La representación de stakeholders presenta los actores y partes involucradas en el desarrollo del componente de extracción y almacenamiento, presentando las tareas que realizan cada uno, a continuación:

| Stakeholder | Tareas |
|---|---|
| Arquitecto de la solución | ✓ Gestionar las prioridades, mantener al equipo del proyecto enfocado en los objetivos. Supervisar el establecimiento de la arquitectura del sistema, planificación, control del proyecto y calidad del software.<br>✓ Seleccionar las tecnologías más adecuadas que se ajusten al desarrollo del componente de extracción. |

| | |
|---|---|
| Analista de requerimientos | ✓ Reconocimiento del problema, levantamiento de los requerimientos funcionales y no funcionales del componente mediante entrevistas u otro medio.<br>✓ Elaboración de la documentación del componente de extracción y almacenamiento.<br>✓ Responsable de realizar los diagramas respectivos de la solución (clases, secuencia, entre otros.) |
| Director de Diseño | ✓ Especificar y validar los requerimientos del sistema, elaboración del modelo de diseño, diseño de pruebas funcionales sobre el sistema.<br>✓ Definir los estándares de programación y diseño de programas, recomendados.<br>✓ Definir los principales flujos de datos entre programas y funciones.<br>✓ Diseñar el esquema de datos lógico y físico.<br>✓ Definir los entornos de hardware y software, proponiendo alternativas.<br>✓ Documentar los diagramas de diseño alternativos, sí existen.<br>✓ Definir los programas y sus principales funciones. |
| Equipo de desarrollo y pruebas | ✓ Implementar la solución teniendo en cuenta los requisitos y requerimientos del sistema. Escribir, depurar y mantener el código del componente.<br>✓ Recoger los progresos y estados del proyecto.<br>✓ Instaurar los procedimientos para recoger tiempos, si resulta apropiado.<br>✓ Obtener la aprobación del plan de trabajo por parte de la dirección.<br>✓ Realizar las pruebas de unidad, hasta que los programas se adapten a las especificaciones descritas en las etapas anteriores<br>✓ Hacer las pruebas y tests del sistema. |

| | |
|---|---|
| Cliente | ✓ Atender a las reuniones o entrevistas con el objetivo de hacer un adecuado levantamiento de requerimientos.<br>✓ Conocer los procesos o comportamiento del contexto en el cual se desarrolla el componente de extracción y almacenamiento. |

Tabla 2. Representación de stakeholders

## 3. DEFINICIONES DE PUNTOS DE VISTA

Para la definición de los puntos de vista de la arquitectura del sistema se toma como referencia el modelo de vistas de arquitectura describiendo en que consiste y los stakeholders asociados. La arquitectura de software se ocupa de la abstracción, la descomposición y la composición, el estilo y la estética. También se ocupa del diseño y la implementación de la estructura de alto nivel del software(Kruchten, 1995). El modelo de arquitectura 4 +1 organiza la descripción de una arquitectura de software utilizando cinco vistas de concurrencia (ver Figura 1), cada una de las cuales aborda un conjunto específico de asuntos. A continuación, se describen las vistas tomadas como referencia (Kruchten, 1995) desde las cuales se modela la arquitectura de software para el componente de extracción y almacenamiento.

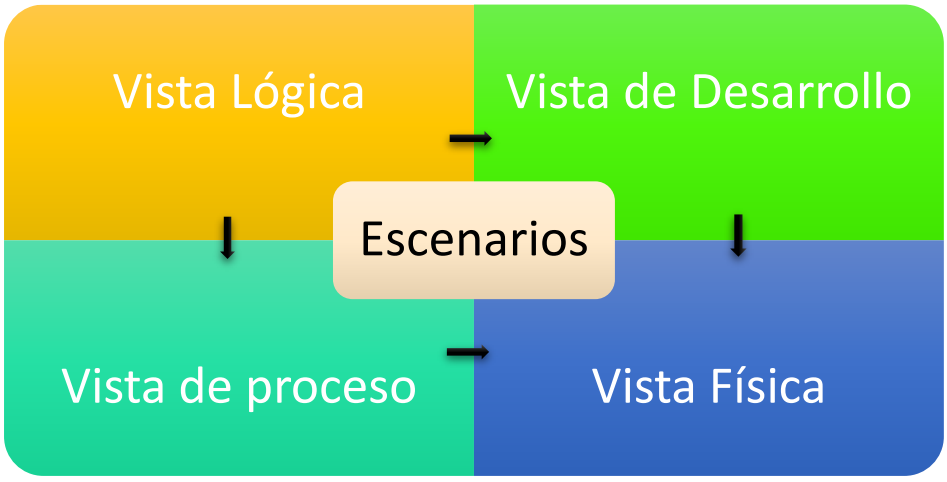



**Figura 1.** Modelo de vistas de arquitectura 4 + 1

## 3.1 Punto de Vista lógico

La vista lógica se encarga principalmente los requisitos funcionales, es decir, los servicios que el sistema debe proporcionar a los usuarios finales. Manejando objetos o clases de objetos que aplican los principios de abstracción, encapsulación y herencia(Tu & Godfrey, 2001). Adicionalmente ayudan al análisis funcional, identificar mecanismos y elementos de diseño que son comunes en todo el sistema.

### 3.1.1 Lenguajes y Stakeholders

Usualmente los diagramas tenidos en cuenta en esta vista son los diagramas de secuencia, comunicación o clase.

Los stakeholders asociados este punto de vista son el cliente y el analista de requerimientos.

## 3.2 Punto de Vista de proceso

La vista del proceso tiene en cuenta algunos requisitos no funcionales. Trata la concurrencia y la distribución, la integridad del sistema y la tolerancia a fallas(Kruchten, 1995). La vista de proceso también especifica qué subproceso de control que ejecuta cada operación de cada clase identificada en la vista lógica.

### 3.2.1 Lenguajes y Stakeholders

Los diagramas tenidos en cuenta en esta vista son los diagramas de actividades.

Los stakeholders asociados este punto de vista son el director de diseño y el arquitecto de la solución.

## 3.3 Punto de Vista físico

La vista física tiene en cuenta los requisitos no funcionales del sistema, como la disponibilidad del sistema, tolerancia a fallos, el rendimiento y la escalabilidad(Tu & Godfrey, 2001). Se pueden usar varias configuraciones físicas diferentes por ejemplo para desarrollo y pruebas y otras para la implementación del sistema en varios nodos o para diferentes clientes. La asignación del software a los nodos debe,

por lo tanto, ser muy flexible y tener un impacto mínimo en el código fuente en sí mismo.

### 3.3.1 Lenguajes y Stakeholders

Los stakeholders asociados este punto de vista son el diseñador y el arquitecto de la solución.

Los diagramas tenidos en cuenta en esta vista son los diagramas despliegue.

## 3.4 Punto de Vista de desarrollo

La vista de desarrollo se centra en la organización de los módulos de software reales en el entorno de desarrollo de software(Tu & Godfrey, 2001). El software está empaquetado en programas o subsistemas que pueden ser desarrollados por uno o más desarrolladores. lustra el sistema de la perspectiva del programador, adicionalmente está enfocado en la administración de los componentes de software

### 3.4.1 Lenguajes y Stakeholders

Los diagramas tenidos en cuenta en esta vista son los diagramas de paquetes, o diagramas de componentes.

Los stakeholders asociados este punto de vista son el arquitecto de la solución y el equipo de desarrollo.

## 3.5 Escenarios y casos de uso

Los escenarios son en cierto sentido una abstracción de los requisitos más importantes. Su diseño se expresa usando diagramas de escenarios de objetos y diagramas de interacción de objetos(Kruchten, 1995). Los escenarios describen secuencias de interacciones entre objetos, y entre procesos.

### 3.5.1 Lenguajes y Stakeholders

Los stakeholders asociados este punto de vista son cliente y el arquitecto de la solución.

Los diagramas tenidos en cuenta en esta vista son los diagramas de secuencia, comunicación o clase.

# 4. REPRESENTACIÓN ARQUITECTONICA

## Diagrama de Arquitectura del Componente

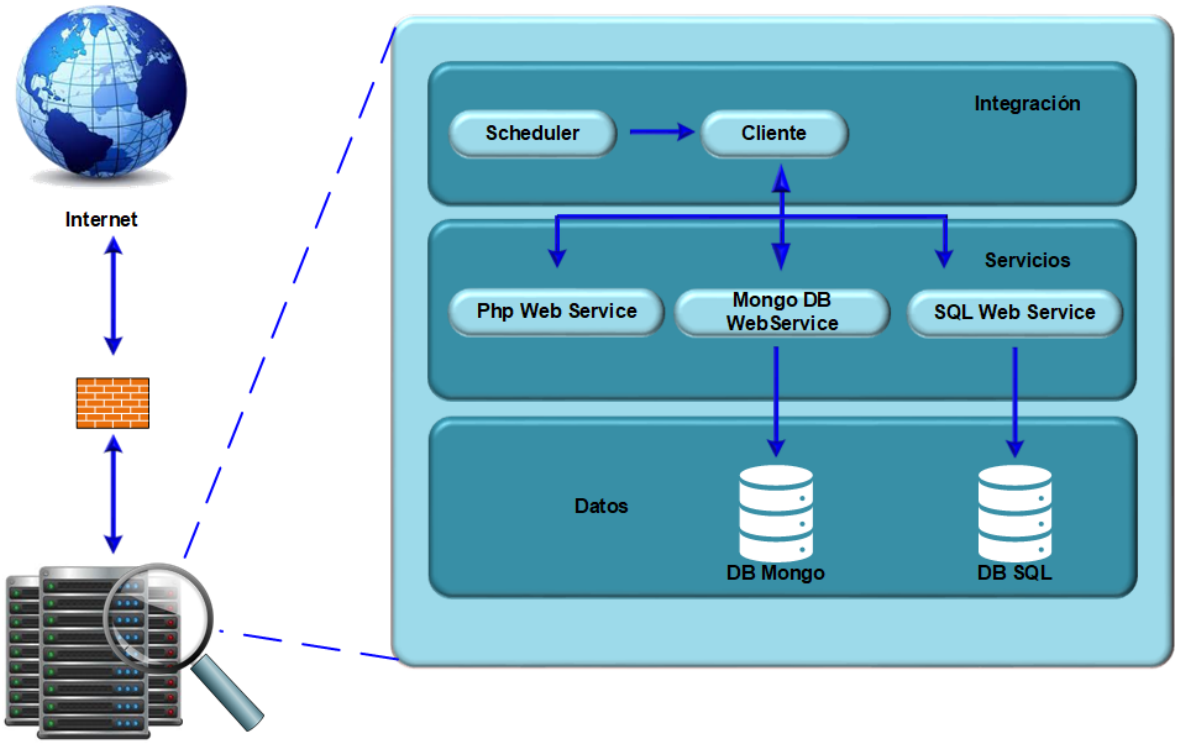



**Figura 2.** Representación arquitectónica del componente de extracción y almacenamiento.

### 4.2 Visión General

Siguiendo los requerimientos definidos en el documento de especificación de requerimientos, la arquitectura del componente de extracción y almacenamiento estará dividida en tres módulos principales que son, módulo de integración, módulo de servicios y módulo de datos (Ver Figura 2.).

En primer lugar, se encuentra el módulo de integración, está compuesto por dos componentes que permiten realizar el control de ejecución de las operaciones que se realizan en los módulos de servicios y datos permitiendo de esta manera la comunicación de los tres módulos a través del módulo de integración. La descripción en detalle de cada componente se hace en la sección 5.4Vista de Desarrollo del presente documento.

En segundo lugar, se encuentra el módulo de servicios, en él se encuentran agrupados los componentes core del desarrollo del proyecto porque son los encargados de realizar los procesos funcionales de extracción y el almacenamiento, este módulo se comunica por medio de interfaces con el módulo de integración y datos.

Finalmente se encuentra el módulo de datos, hace referencia a las bases de datos que se van a implementar en el desarrollo de la solución.

Dadas las condiciones anteriores, la funcionalidad del sistema sería de la siguiente manera, la capa de integración consume a la capa de servicios y controla el hilo de ejecución de cada servicio, en este punto se hacen los llamados a los servicios de parametrización, extracción y almacenamiento.

El orden de ejecución está dado en detalle en la vista de proceso del componente.

Por otra parte, el hecho que varios componentes hagan parte de un mismo modulo no significa que estén alojados físicamente en la misma máquina, entendiendo como máquina a un servidor o equipo físico. De esta manera las capas o módulos del componente se pueden distribuir en diferentes puntos o nodos, ahora bien, la descripción en detalle de esta característica se encuentra en la sección 5.3Vista Física del presente documento.

# 5. VISTAS

## 5.1 Vista Lógica

Para describir el funcionamiento general del componente se emplea la vista de lógica que es la encargada de describir y representar los requerimientos funcionales del sistema, para el caso particular se utiliza el diagrama de secuencia, a continuación:

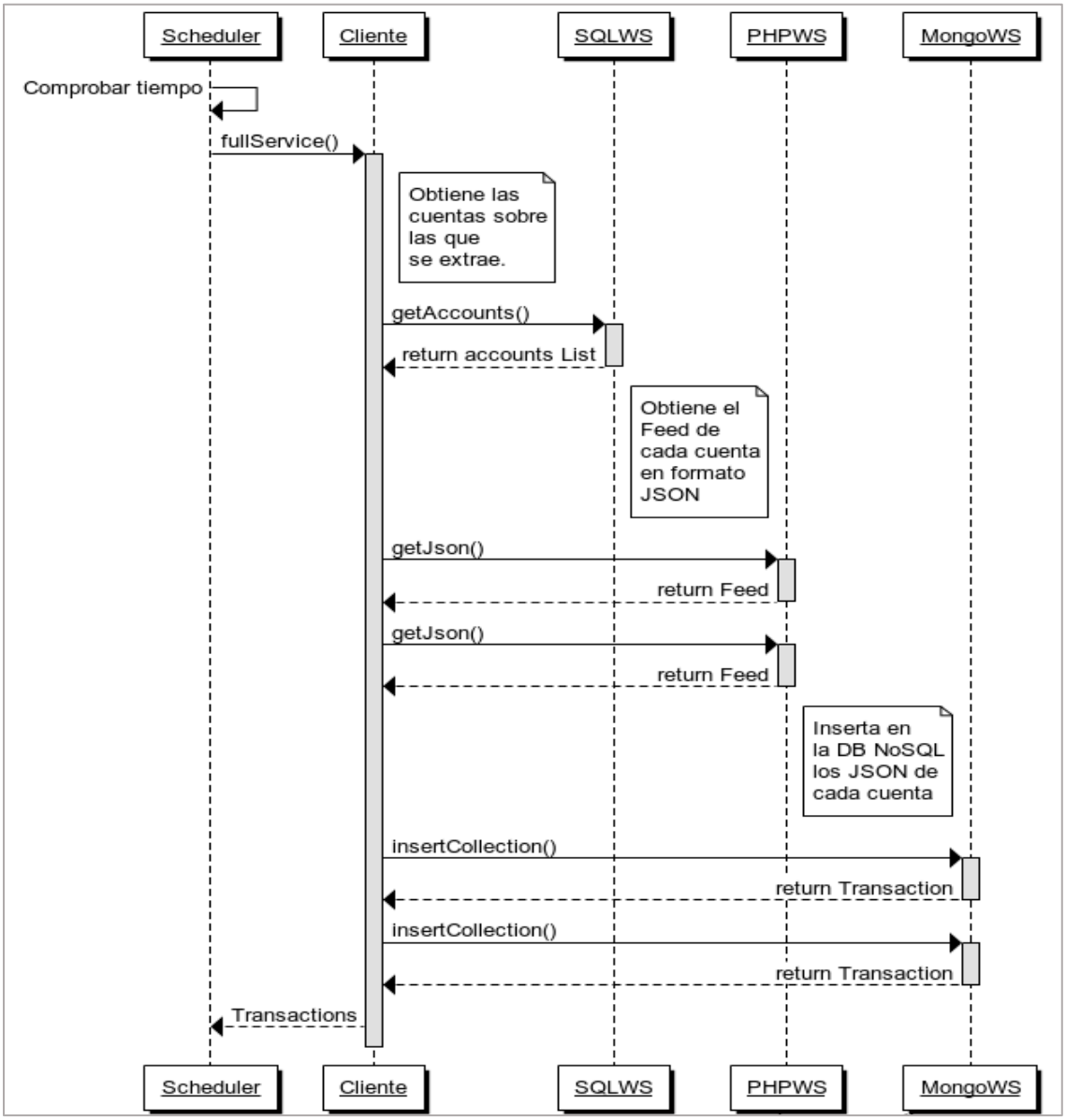



**Ilustración 3.** Diagrama de Secuencia

## 5.2 Vista de Proceso

La vista de proceso especifica los proceso y subprocesos de control que ejecuta cada operación de cada funcionalidad identificada en la vista lógica.

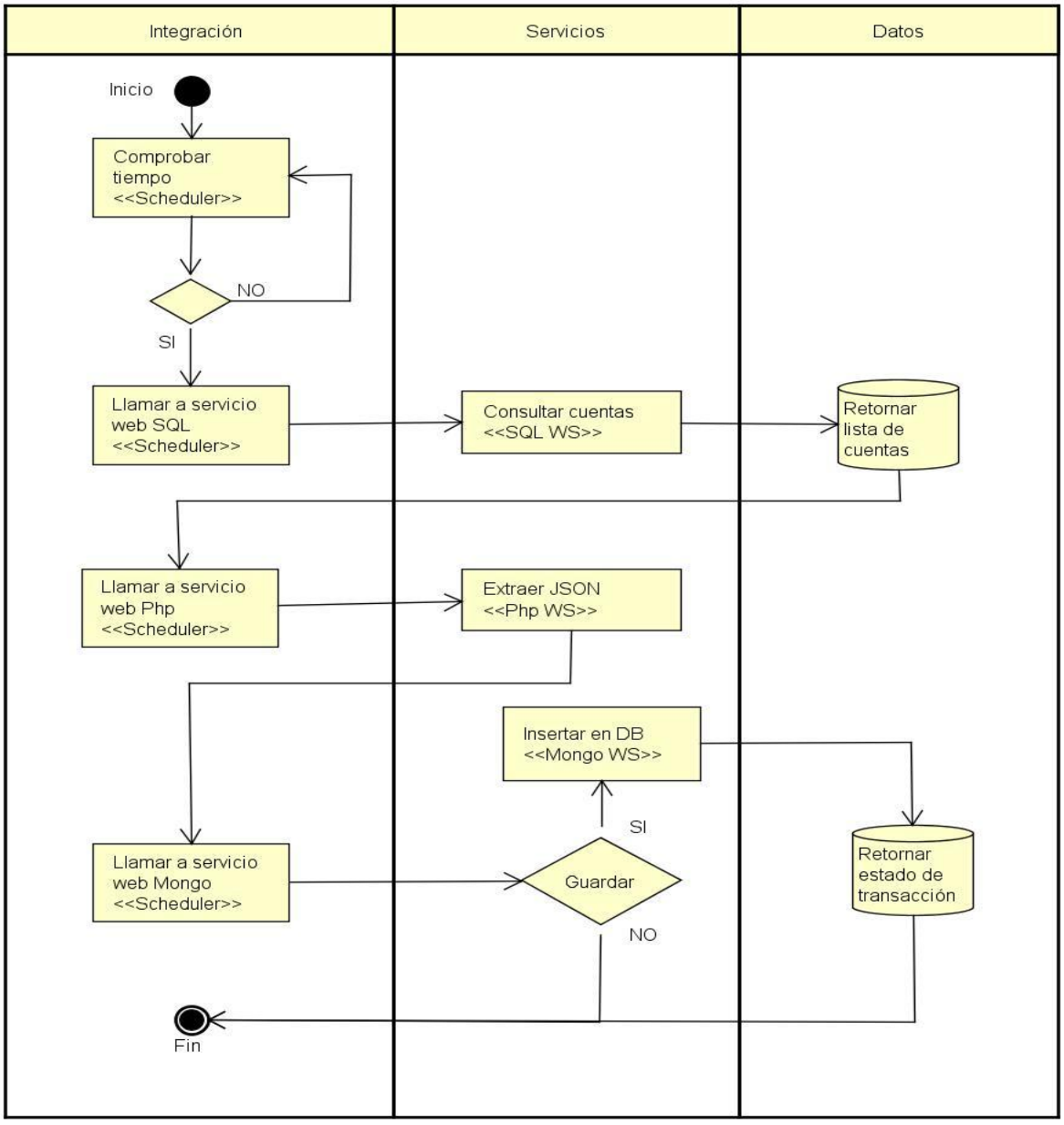



**Ilustración 4.** Diagrama de Actividades del componente. Fuente: Elaboración propia.

## 5.3 Vista Física

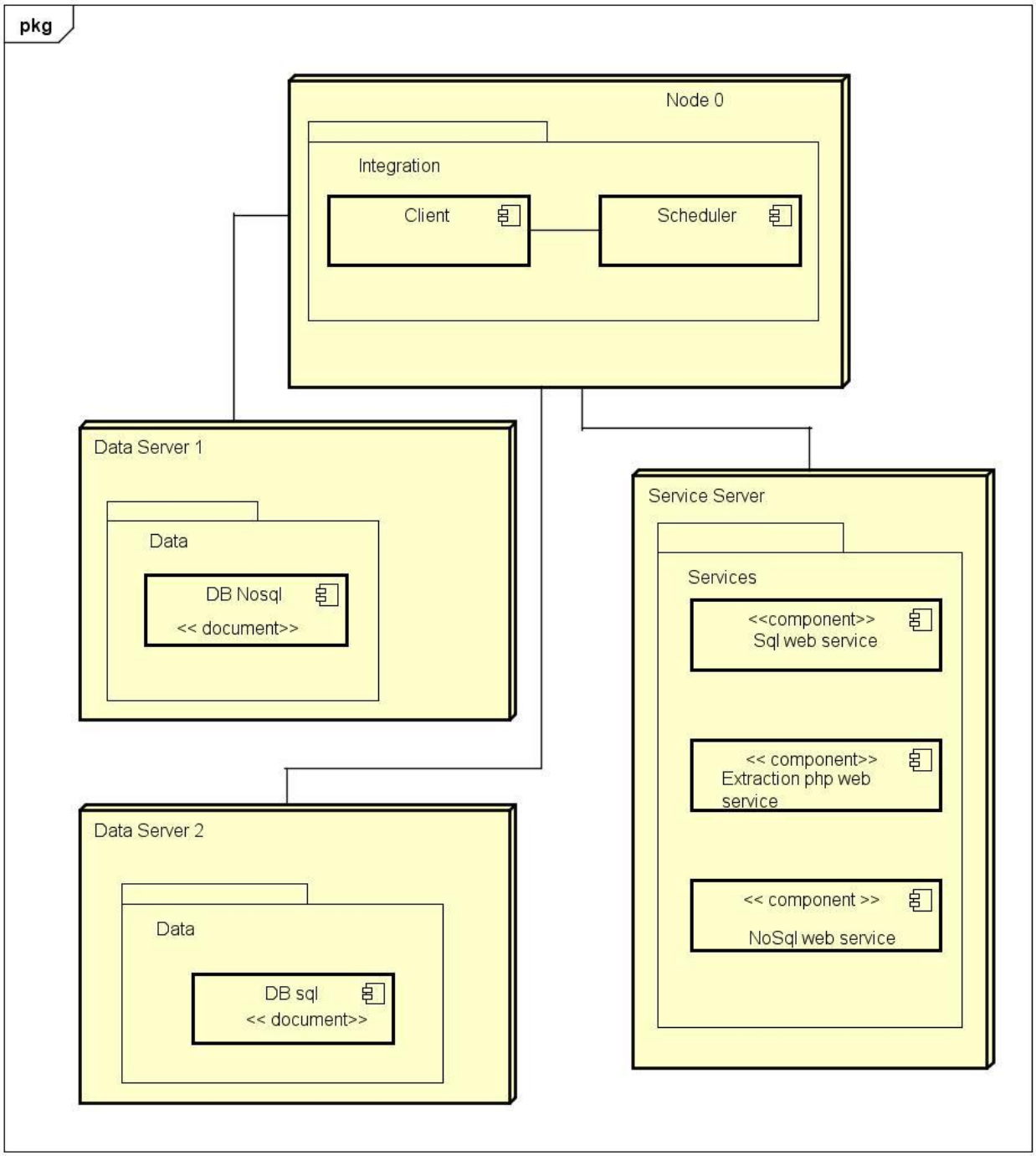



**Ilustración 5.** Diagrama de despliegue.

El diagrama de despliegue permite ilustrar cómo funciona el sistema físicamente, a continuación, se realiza una breve descripción de las características óptimas para cada máquina o servidor para realizar la instalación del sistema.

| Nombre | Nodo de despliegue 0 |
|---|---|
| Especificación | ✓ Disco duro de 1 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 2Gz o superior. |
| Capas | Integración |

*Tabla 3. Especificación de Nodo de Despliegue*

| Nombre | Data Server 1 |
|---|---|
| Especificación | ✓ Disco duro de 10 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Apache HTTP Server<br>✓ Memoria RAM 16 Gb o superior |
| Capas | Datos |

*Tabla 4. Especificación de Data Server 1*

| Nombre | Data Server 2 |
|---|---|
| Especificación | ✓ Disco duro de 10 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Memoria RAM 16 Gb o superior<br>✓ Apache Tomcat v7 |
| Capas | Datos |

*Tabla 5. Especificación de Data Server 2*

| Nombre | Servicie Server |
|---|---|
| Especificación | ✓ Disco duro de 1 Tera, 89.16 MB para el JDK versión 6.0 o superior.<br>✓ Procesador Intel Core o procesadores compatibles a 4Gz o superior.<br>✓ Memoria RAM 16 Gb o superior |
| Capas | Servicios |

*Tabla 6. Especificación de Service Server*

## 5.4 Vista de Desarrollo

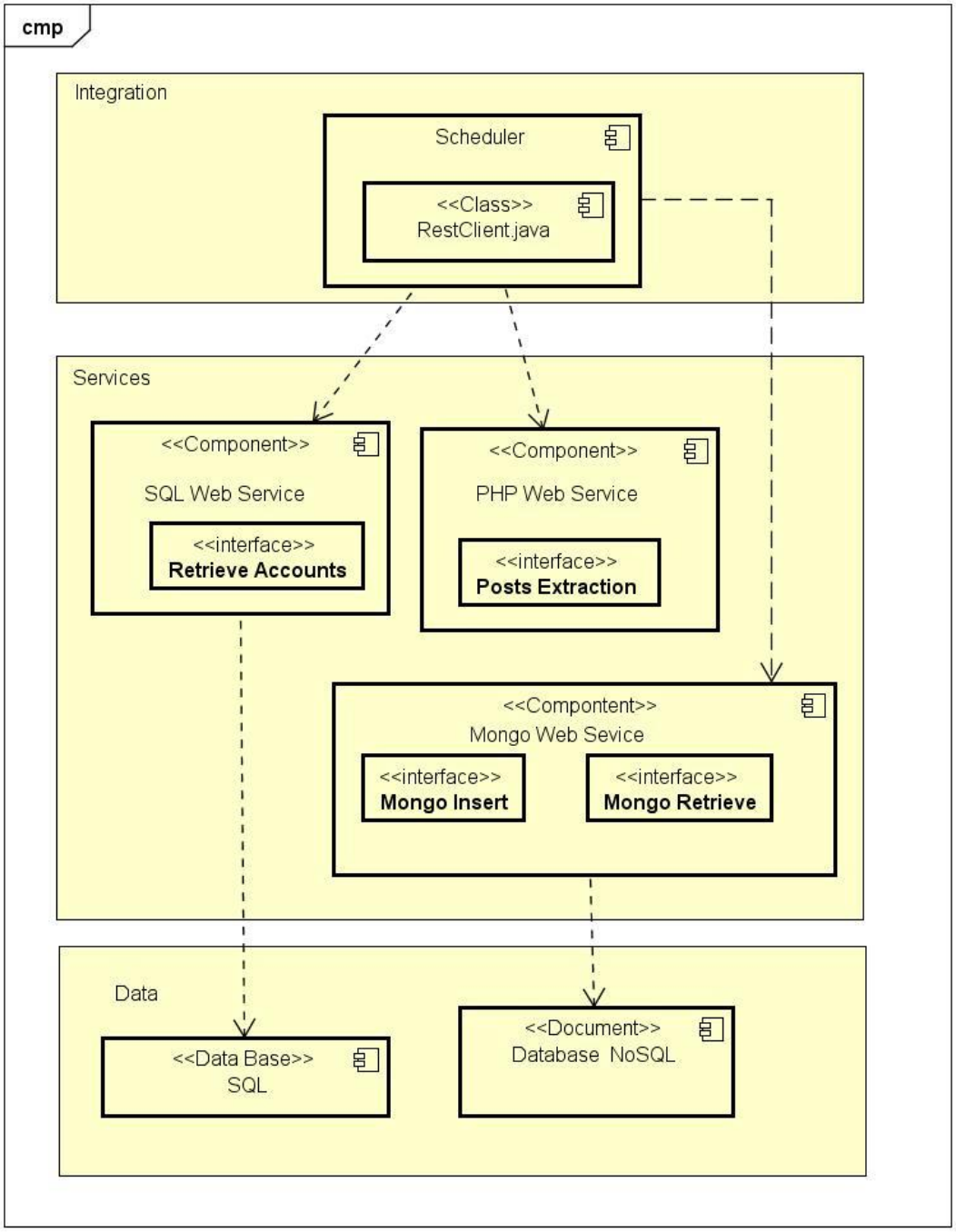



**Ilustración 6.** Diagrama de componentes.

*Scheduler:* Este componente es el encargado de integrar toda la funcionalidad del "Componente de extracción y almacenamiento", interactuando con los servicios web, permitiendo la ejecución de manera ordenada de los diferentes componentes que hacen parte del sistema.

*SQL Web Service:* Es el componente encargado de consultar las llaves de autenticación y las cuentas sobre las cuales se quiere extraer información, esta información será utilizada respectivamente como las entradas del componente de extracción php.

*PHP Web Service:* Además de comunicarse con el Scheduler, este componente puede considerarse como el corazón del sistema debido a su importancia, es el encargado de la extracción de las publicaciones o posts de la red social Facebook. Por medio del API(Imielinski, Virmani, & Abdulghani, 1996) "Graph Api" y de las credenciales de acceso obtenidas por el componente SQL extrae los datos de las cuentas requeridas en formato Json y los retorna para que puedan ser almacenados respectivamente.

*Mongo Web Service:* Se encarga de gestionar el almacenamiento de la información entrante por parte del componente de extracción Php y consultar la información contenida en la base de datos NoSQL, por medio de dos interfaces llamadas Mongo Insert y Mongo Retrieve, ambas interfaces intercambian información en formato Json.

*SQL Database:* Este componente se encuentra en la capa de datos de la arquitectura del componente general, se encarga de almacenar las llaves de acceso al API de extracción, de igual manera, contiene las cuentas de las páginas de las que se extraerán los posts o publicaciones.

*NoSQL Database:* Este componente al igual que el anterior, se encuentra en la capa de datos, es el encargado de almacenar toda la información extraída por parte del componente Php, es NoSQL por la ventaja en el almacenamiento de grandes cantidades de información en comparación a una base de datos relacional.

## Glosario

Apache Tomcat: Es un servidor de aplicaciones libre, que implementa las tecnologías desarrolladas en la plataforma Java EE y permite ejecutar aplicaciones desarrolladas bajo este lenguaje.

HTTP: El protocolo de transferencia de hipertexto es el protocolo usado en cada transacción que se ejecuta en la web. Java EE Es una plataforma de programación para desarrollar y ejecutar aplicaciones en lenguaje de programación java. Está enfocada para aplicaciones con gran cantidad de transacciones y/o empresariales.

TCP : Es uno de los principales protocolos de internet. Se pueden usar conexiones TCP para crear conexiones entre dos nodos para enviar un flujo de datos.

UML : Es el lenguaje de modelado de sistemas más conocido y utilizado en el mundo.

WebService: Es una tecnología que utiliza intercambio de mensajes XML o JSON y que permite a dispositivos externos a un sistema, acceder a él sin necesidad de implementaciones adicionales.